

複合数学 Web サービスに関する研究

中西 智美
TOMOMI NAKANISHI *

池田 理世
Riyo IKEDA †

甲斐 博
HIROSHI KAI ‡

愛媛大学大学院理工学研究科

GRADUATE SCHOOL OF SCIENCE AND ENGINEERING, EHIME UNIVERSITY

1 はじめに

低コストでより柔軟な PSE 構築フレームワークとして Lupin[1] が提案されている。特に数学分野を対象とした Web サービスによる Lupin の実装は数学 Web サービスと呼ばれる。数学 Web サービスを工学や教育に応用するためには、数式処理や数値計算やグラフ描画などを行う個々の既存の Web サービスを組み合わせて新しい Web サービスを構築する機能が必要になる。ビジネス分野では、そのような目的のために Web Services Business Process Execution Language (BPEL) [9] が提案されている。そこで本研究では、BPEL により数学 Web サービスを連携した複合数学 Web サービスの検討を行う。また、複合数学 Web サービスの数学教育への応用を行う。

2 Lupin

PSE とは、対象とする問題領域において必要となる全ての計算機資源を提供するシステムである [6]。PSE の特徴として高度な問題解決方法の提供、解法の自動または半自動選択機能、新しい解法を容易に組み込める事などがあげられる。

Lupin のアーキテクチャを図 1 に示す。Lupin では、Web からアクセスできる計算機資源を Lupin サービスと呼び、Lupin サービス生成機構、Lupin 発見機構、PSE 構成機構から Lupin は構成される。Lupin サービス生成機構は、サービスの作成、配置を行い、そのサービス情報を Lupin サービス発見機構に登録する。Lupin 発見機構はサービスの登録、検索を行う。PSE 構成機構は必要なサービス情報を Lupin 発見機構から検索し、サービスを利用することにより PSE を構築する。以下これら 3 つの機能の実装について述べる。

- Lupin サービス生成機構における Web サービスの呼び出しは SOAP を使って行われる。現在、数学 Web サービスとして、数式処理サービスや、数値計算サービス、グラフ表示サービス等を開発している [2, 3, 4]。Web サービスでは通常 WSDL がサービス登録に用いられるが数学的情報を記述できない。そこで、我々は数学的情報を記述するために、OWL-S を用いることを提案している [5]。

*nakanishi@hpc.cs.ehime-u.ac.jp

†iked@hpc.cs.ehime-u.ac.jp

‡kai@cs.ehime-u.ac.jp

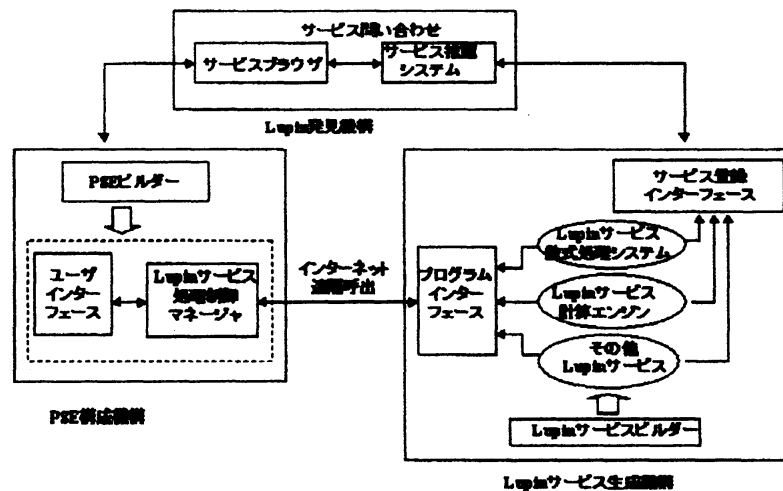


図 1: Lupin のアーキテクチャ

- Lupin サービス発見機構の実装は UDDI を用いて行うことができる。問題を与えるとそれを解決するサービスを検索する。問題の記述は MathML や OpenMath を用いて行う。より高度な推論を行うため、MONET や GAMS で開発された数学オントロジと登録されたサービスとの関連づけを行っている。これにより、親子関係にあるサービスの検索が可能になる [5]。
- PSE 構成機構は数学 Web サービスのクライアントにあたる。これまで Web アプリケーションを利用して実装してきたが、対話的な処理には向いていない。これを Ajax や xfy[10] などを用いた技術に置き換えていくことを検討している。PSE ビルダーは、サービスを結合し PSE を構築するものであるが、その実装が本研究の目的であり、次節で検討する。

3 BPEL による複合数学 Web サービス

3.1 BPEL の数学 Web サービスへの応用

BPEL は WSDL などの Web サービスの標準技術を用いてビジネスプロセスを記述するための言語である。BPEL プロセスは、基本的にはアルゴリズムをフローチャートに似た形で表現したものであり、`process` をルート要素とし、プロセスにおける各ステップをアクティビティと呼ばれる子要素で定義した XML 文書となる。そしてその基本アクティビティを構造化アクティビティで結合することにより、複雑なプロセスフローを構築することができる。表 1 に主要なアクティビティを示す。

例 1

BPEL プロセスを用いた数学 Web サービスの例として GCD 計算について考える。Risa/Asir には以下の 2 つの多項式 GCD を求める命令が存在する。

- `gcd(p,q)` ... p, q を有理数体上の多項式としての GCD を返す。結果は整数係数で、かつ係数の GCD が 1 になるような多項式、または、互いに素の場合は 1 を返す。
- `gcdz(p,q)` ... p, q がともに整数係数の場合、整数環上の多項式としての GCD を返す。すなわち、`gcd()` の値に、係数全体の整数 GCD の値を掛けたものを返す。

表 1: アクティビティの例

	名称	説明
基本アクティビティ	invoke	サービス呼び出す
	receive	ユーザからのメッセージを受け取る
	reply	入出力操作の応答を生成する
	assign	データの作成, または変更を行う
構造化アクティビティ	sequence	順番に実行するアクティビティを定義する
	switch	条件分岐を実行する
	while	ループを定義する
	flow	並列に実行できるアクティビティを定義する
	links	並列的に実行されるアクティビティの中で, links を使用して実行順序の制約を示すことができる

この 2 つの命令はいずれも 2 つの入力多項式に対する GCD を求める命令であるが, どちらの命令で実行するかによって求められる解が異なる. Lupin サービス検索機構には, **gcd()** を実行するサービスを **AsirGcdService** と, **gcdz()** を実行するサービスを **AsirGcdzService** が登録されている. GCD を UDDI で検索すると図 2 のようなサービスリストが返される. ユーザはこれらから望ましいサービスを判断することになる. ここでは BEPL を利用した数学 Web サービスの一例として, 与える多項式の型でどちらのサー

Service Name	Asir Gcdz Service
Actor Name	LupinProvider
Service Description	Gcd service
WSDL URI	http://100.0.234.9090/axis/services/AsirGcdzService?wsdl
Service Name	Asir Gcd Service
Actor Name	LupinProvider
Service Description	Gcd service
WSDL URI	http://100.0.234.9090/axis/services/AsirGcdService?wsdl

図 2: 検索結果

ビスが適切か自動判断するサービスを考える.

BPEL プロセスに対する入力を, 2 つの多項式とユーザの指定する係数の型とする. このとき, 次のような分岐により, 適切なサービスを判定する.

- p, q が有理数係数の多項式の場合, **AsirGcdService** を選択する.
- p, q が整数係数の多項式の場合, ユーザの指定した p, q の係数の型に従って **AsirGcdService** または **AsirGcdzService** を選択する.

結果は適切なサービスの URI とする. この Web サービスを **CategorizeService** と呼ぶ. 構築した BPEL プロセスの内部ビューを図 3 に示す. 図 3 における BPEL プロセスの動作は以下のような手順になる.

1. **receive** アクティビティによってクライアントからの入力を受け取り, その情報を **request** 変数に保管する.
2. **invoke** アクティビティによって **CategorizeService** を呼び出す.

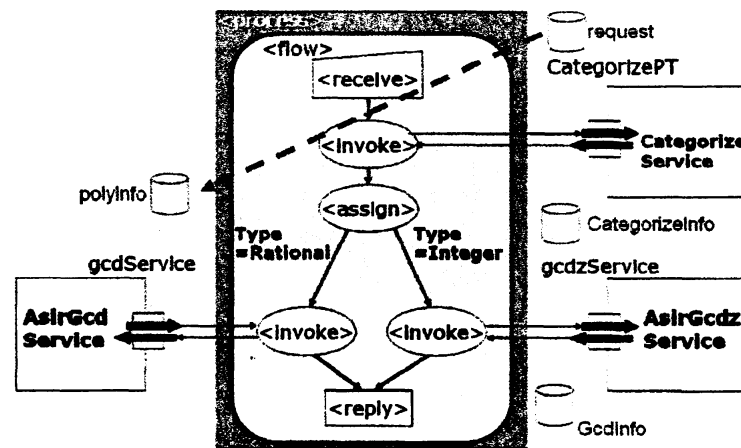


図 3: 複合 Web サービスのプロセス内部ビュー

3. CategorizeService は request の情報を入力としてどちらのサービスを使うか判別し、その出力を categorizeInfo に保管する。
4. assign アクティビティによって request 変数から 2 つの多項式の情報のみを取り出し、polyInfo 変数に保管する。この多項式の情報 OpenMath 形式である。
5. categorizeInfo の情報に適したどちらかの invoke によって、AsirGcdService (または AsirGcdzService) が呼び出される。
6. AsirGcdService (または AsirGcdzService) は polyInfo の情報を入力として GCD を計算し、その結果を出力として GcdInfo に保管する。
7. reply によって GcdInfo の情報が複合 Web サービスの出力としてクライアントに返される。出力は OpenMath 形式で返される。

図 4: 複合 GCD 計算サービスの入出力画面

JSP により作成したサービス用実行クライアントを示す (図 4)。まずフレーム左の入力画面より GCD を求める 2 つの多項式を入力し、次に計算する GCD の係数型を指定する。インターフェースではユーザの

操作を容易にするために、入出力の操作は MathML の表示・入力支援ソフトウェアである WebEQ を用いる。MathML と OpenMath の相互変換は XSLT を用いた。実行例は 2 つの多項式 $\frac{5}{19}x + 5y$ と $\frac{15}{19}x + 15y$ の GCD を計算している。このとき係数の型指定を Integer としているが、入力した多項式が有理数係数と判断されたため型指定は無視され、AsirGcdService を使って有理数係数として計算されている。 $x + 19y$ が結果として得られている。

このように BPEL を用いると Web サービスを関数のように考えてプログラミングできる。

3.2 数学 Web サービスにおける BPEL の問題点

例 1 から BPEL を用いて数学 Web サービスを連携するには、以下の問題点があることが分かる。

- BPEL プロセスにおいて、各サービスの出力は変数として保管され、また各サービスは変数を入力として受け取る。しかし例 1 のステップ 4 にあるように、変数に 2 つの数式が保持されておりサービスの入力はそのうちの 1 つのみであった場合、1 度別の変数に必要な数式を取り出してからあらたに変数を定義し、それをサービスの入力としなくてはならない。変数間のコピーでは変数の一部のみをコピーできるが、サービスを呼び出す `invoke` アクティビティでは変数の一部を直接参照することが出来ないため、プロセスの構造が複雑になる。
- 数学 Web サービスのインターフェース記述として OWL-S を用いている。しかし BPEL では、WSDL と BPEL 中のプロセス定義が制御フローを形成し、作成した BPEL プロセスは WSDL を使って再び外部に公開される。つまり OWL-S のみでは BPEL プロセスのインターフェース記述を行うことは出来ない。よって BPEL プロセスを構築する場合、現状では、検出された連携するサービスの OWL-S から WSDL を取得し、BPEL プロセスのインターフェース記述をしなければならない。

また、もし連携したい既存のサービスに異なるバインディング形式のサービスが混在していた場合、プロセス中で何らかの方法を用いて型変換を行わなくてはならないということも注意しなくてはならない。Web サービスのバインディング方式には、RPC 形式と Document 形式が存在する。ビジネスプロセスとしての BPEL プロセスでは、RPC 形式を使ったメソッドの引数と戻り値のデータのみを交換するのが一般的である。しかし数学 Web サービスにおける数式のメッセージ交換は、OpenMath や MathML などの XML 形式で行われる。つまり数式情報を保持するために、数学 Web サービスでは Document 形式でデータの交換を行う。これらのサービスの混在はプロセスを複雑にする。

いずれも BPEL の仕様における問題であるが、複合数学 Web サービスを容易に作成できるように拡張を今後検討する必要がある。

4 複合数学 Web サービスの応用

Lupin を実際に工学や数学教育へ応用する一例として関数描画のためのグラフ表示サービスの応用があげられる [3, 4]。

しかし、グラフ表示サービスを不定積分の表示を行う教育コンテンツに利用する場合を考えると、従来ではグラフ表示サービスに機能を追加するしかなかった。そのためだけに積分の計算機能をグラフ表示サービスに組み込むこともできるが、余分なコストがかかる。本研究で示した方法を用いることにより、既存の数学 Web サービスを再利用できる。

例えば、不定積分

$$\int_0^1 \frac{1}{(x+1)(x+2)(x+3)+0.1} dx$$

の関数形を表示することを考える。被積分関数をハイブリッド積分を行う数学 Web サービスへの入力とした結果、内部的に次式が不定積分として得られる。

$$-1.0131 \log(x + 1.8989) + 0.4373 \log(x + 3.40466) + 0.5942 \log(x + 1.0543)$$

この結果を BPEL による連携によりグラフ表示サービスに送ると図 2 のようになる。すなわち、複合数学 Web サービスを用いることにより様々なグラフを表示する教育コンテンツを比較的容易に作成できるようになる。

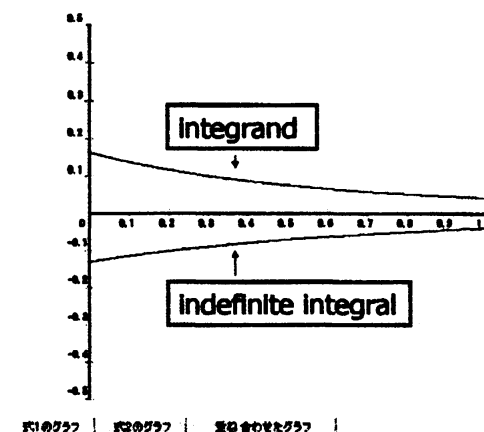


図 2 数学 Web サービスによる不定積分の表示

5 おわりに

本研究では、BPEL により数学 Web サービスの連携ができることを示した。これを発展させることにより複雑な PSE を構築することができると考えられる。但し、BPEL では OWL-S への対応が不十分であったり、数学 Web サービスで必要とされるようなデータ交換が簡単にできないためプロセスが複雑になるという問題が残る。このための改良については今後の課題である。

参 考 文 献

- [1] Kai Li, Masato Sakai, Yukihiro Morizane, Masahiro Kono and Matu-Tarow Noda, : Lupin : towards the framework of web-based problem solving environments, Proceedings of the 8th Asian Technology Conference in Mathematics, Taiwan, 2003, pp.276-285.
- [2] Hiroshi Kai : A plot web service using approximate algebra, Proceedings of The Tenth International Conference on Application of Computer Algebra, Beaumont, Texas, USA, 2004, p.161.
- [3] Takumi Watanabe, Ikko Kadota, Yukihiro Morizane, Masahiro Kono, Li Kai, Hiroshi Kai and Matu-Tarow Noda : An implementation of Lupin's plot web service, ISSAC2004, <http://www.acm.org/sigsam/issac/2004/>, 2004, pp.1-3.
- [4] 渡邊巧, 甲斐博, 野田松太郎 : 数学 Web サービスにおけるグラフ表示サービスとその教育への応用, 数理解析研究所講究録 1456, 2005, pp.77-86.

- [5] 中西智美, 池田理世, 甲斐博 : 数学 Web サービスの実装とプロセス実行に関する研究, 信学技報, vol.106, no.327, KBSE2006-31, 2006, pp.37-42.
- [6] Eliaa N. Houstis and John R. Rice : On the Future of Problem Solving Environments, <http://www.cs.purdue.edu/people/jrr>, 2000, pp.1-78.
- [7] MONET, <http://monet.nag.co.uk/cocoon/monet/>
- [8] GAMS, <http://gams.nist.gov>
- [9] WSBPEL, <http://www.oasis-open.org/>
- [10] xfy, <https://www.xfytec.com/>