

# Ruby で Maple を動かすためのインターフェースの開発

情報科学科 西谷研究室 3528 村瀬愛理

## 1 研究目的

Ruby では数値計算のライブラリ開発が遅れており，Ruby 上では高等な関数（素数を求めたり，最小公倍数を求めるなど）を使った数式処理を行うのが難しい．また，Ruby 以外の数式処理ソフトウェアを別に立ち上げて別々で作業したり，慣れない別の言語を勉強し直したりするよりも，Ruby のみでプログラミングする方が開発速度の格段の向上が期待できる．そこで本研究では，Maple を Ruby 上で呼び出し，Maple に高等な関数や桁数の大きな数値を用いた計算をさせて，その結果を Ruby が取得するインターフェースライブラリを開発を目的とする．

## 2 手法

**Maple との連携** Maple は一般的に，グラフや数式の綺麗な出力や，数式の入力を初心者が直感的におこなえるように Java で作られた GUI を使って実行する．それとは別に command line で実行される計算エンジン部が用意されている．ここで開発する Ruby ライブラリでは，このエンジンに直接働きかけて操作する．Ruby で外部コマンドを実行する gem library の systemu を使って，出力を得るようにしている．Ruby code で要求コードを受け取った場合，そのコードを tmp ファイルに書き込む．それを Maple で実行し，結果をテキストファイルで受けとり出力を得る．

**Maple 関数の類型化** 今回，数多く存在する Maple の数学関数の中から整数論と行列に関するものを選抜し実装した後，入出力に関して類型化し，それぞれの出力に応じて wrapper を作った．表 1 は類型化の一部である．

表 1 実装した整数論に関する関数と入出力.

関数名	振る舞い	入力型	出力型
nextprime	次の素数を求める	int	int
lcm	最小公倍数	int, int	int
gcd	最大公約数	int, int	int
isprime	素数判定	int	boolean

## 3 基本動作

入力された値の次の素数を出力する nextprime を例に説明する．

- mapleruby を require した上で使いたい関数を呼び出す．  
RMaple.new.hogehoge の hogehoge に使いたい関数名（ここでは nextprime）を入れる．
- RMaple クラス内の nextprime 関数が呼び出される．そ

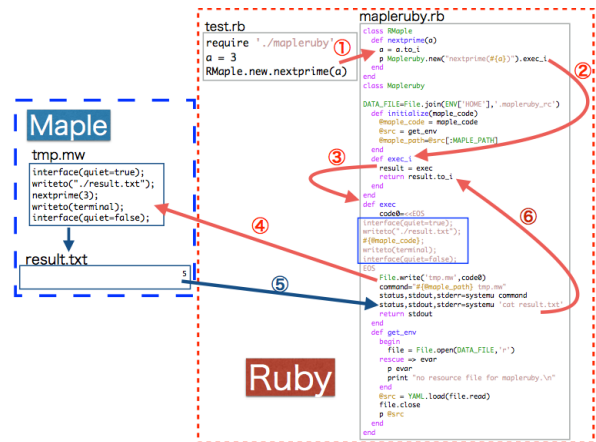


図 1 mapleruby の基本動作.

の後，Mapleruby クラスの exec.i 関数へ”nextprime(3)”が出力される．この出力された文字列がそのまま Maple での計算に使われる．

- 出力された文字列をさらに exec 関数へ出力する．
- 青四角内の内容を Maple へと出力する．この時 `#{@maple_code};` に先ほどの”nextprime(3)”が入る．青四角の内容が Maple に出力・実行されることで得られた答えが result.txt に出力される．
- result.txt に出力された内容を Ruby 側で受け取り，exec.i に再び返す．
- 返された値を to\_i することで int 型に直して解を出力する．

## 4 本研究による成果

本研究で得られた成果をまとめると以下の通りである．

- RSA 暗号化における計算が，Ruby 単体で行うよりも約 100 倍の数まで扱えるようになった．
- 3 行 3 列の行列について逆行列や固有値，固有ベクトルを求めることができた．

従来では，Maple と Ruby を行き来しなければならなかったような計算でも，mapleruby を用いることで Ruby 上のみで完結させられるようになった．

## 5 総括

Ruby だけで桁数の大きい計算や複雑な数学関数を必要とする計算が完結させられることが可能になった．今後は新たな関数の追加や，Maple の特徴である綺麗なグラフ描画ができることを生かしたグラフの出力を可能にするなどが課題として挙げられる．