Resolução de sistemas lineares em $\mathbb{Z}_p$

Taina da Silva
Luiz-Rafael Santos
Luiz Fernando Bossa

Universidade Federal de Santa Catarina
Blumenau, SC, Brasil

tainasilva3012@gmail.com, l.r.santos@ufsc.br,
l.f.bossa@ufsc.br

Resumo

Neste trabalho buscou-se explorar e compreender o funcionamento métodos para resolução de sistemas lineares, nos quais as componentes que definem as equações lineares pertencem ao corpo $\mathbb{Z}_p$, bem como implementá-los computacionalmente. Para isto, exploramos o algoritmo da divisão euclidiana, e definimos e caracterizaremos o conjunto $\mathbb{Z}_p$ como um corpo. Além disso, caracterizamos sistemas lineares no corpo dos números reais, e analisamos algoritmos para resolução de sistemas lineares em $\mathbb{Z}_p$. Finalmente, construímos uma representação computacional de $\mathbb{Z}_p$ por meio da linguagem de programação `Julia`, implementamos os algoritmos para resolução de sistemas lineares em $\mathbb{Z}_p$, e apresentamos uma aplicação em criptografia utilizando cifra de Hill.

Palavras-Chave: Sistemas lineares; Corpo $\mathbb{Z}_p$; Linguagem Julia; Cifra de Hill.

Cifra de Hill

- Criado por Lester S. Hill em 1929;
- Substituição poli-alfabética:
  - tabela de conversão;
  - $\mathbb{Z}_n$ – Inteiros módulo n.
- Baseado em transformações matriciais:
  - Matriz inversível;
  - Elementos de $\mathbb{Z}_n$ que possuam inverso.

Teorema 1. *O conjunto $\mathbb{Z}_p$ é um corpo se, e somente se, p é primo.*

Problema

Implementamos a cifra de Hill para criptografar uma mensagem. Para realizar a criptografia precisa-se de uma chave, a qual precisa ser uma matriz inversível em $\mathbb{Z}_p$ pois para ler a mensagem original precisamos de sua inversa.

- Alguns métodos para calcular a inversa A^{-1} possuem alto custo computacional;
- O método mais barato é resolver n sistemas lineares do tipo $Av_i = e_i$, em que A é a matriz que desejamos inverter, v_i é a i-ésima coluna da inversa e e_i representa a i-ésima coluna da identidade.
- Existem muitos métodos destinados a resolver sistemas lineares do tipo $Ax = b$. Analisamos o comportamento destes métodos no corpo $\mathbb{Z}_p$.
- Escolhemos analisar neste estudo a fatoração LU.

Resultados

Durante o desenvolvimento deste trabalho, nos familiarizamos com conceitos importantes sobre algoritmo euclidiano estendido, realizamos a construção do corpo dos inteiros módulo p, relembramos conceitos importantes acerca de matrizes e sistemas lineares além de métodos utilizados para resolvê-los, estudamos os algoritmos para resolução de sistemas lineares no corpo $\mathbb{Z}_p$ e estudamos a linguagem de programação escolhida além de realizar as implementações e, por fim, apresentamos um exemplo resolvendo uma cifra de Hill.

Número de soluções de um sistema linear em $\mathbb{Z}_p$

O principal resultado teórico que estudamos no trabalho mostra que o número de soluções de um sistema linear no corpo $\mathbb{Z}_p$ é sempre finito, ao contrário do que acontece em um corpo como $\mathbb{R}$.

Teorema 2. *Seja A uma matriz $m \times n$ com elementos em $\mathbb{Z}_p$. Qualquer sistema de equações lineares consistente, com matriz dos coeficientes igual a A, tem exatamente $p^{n-\text{posto}(A)}$ soluções sobre $\mathbb{Z}_p$.*

Construindo um corpo finito em `Julia`

Para realizar os experimentos computacionais, procedemos com a construção de um pacote dentro da linguagem `Julia` que permite operações no corpo $\mathbb{Z}_p$. Para isso, utilizamos a característica *multiple-dispatch* da linguagem escolhida e um construtor do tipo `struct`. A seguir temos o código utilizado para construir a variável base do pacote.

Código 1 Código em `Julia` do construtor do corpo $\mathbb{Z}_p$

```julia
struct IntZp <: Number
    n::Int
    p::Int
    function IntZp(n, p)
        a = n % p
    if a < 0
        a += p
    end
        return new(a, p)
    end
end
```

Também exportamos as diversas operações em $\mathbb{Z}_p$, como soma, subtração, multiplicação, divisão, etc. Assim, se tivermos duas variáveis `a,b` do tipo `IntZp`, então `a+b`, `a-b`, `a*b` e `a/b` nos devolvem variáveis do tipo `IntZp`. É possível acessar a lista completa de códigos utilizados em `https://github.com/TainaSilva3012/IntZp.jl` ou através de

Implementação de fatoração LU e aplicação

Além da construção das operações básicas do corpo $\mathbb{Z}_p$ em `Julia` foi realizada a implementação da fatoração LU neste corpo. Também fizemos a aplicação deste pacote na solução de uma cifra de Hill em $\mathbb{Z}_{113}$, na qual utilizamos a fatoração LU para inverter a matriz e resolver os n sistemas lineares necessários para tal fim. Para testar o pacote, criptografamos em um exemplar de um livro de José Saramago. Esta apresentação tem como base o Trabalho de Conclusão de Curso da primeira autora [6] que pode ser acessado em

Referências

[1] Jeff Bezanson et al. «Julia: A Fresh Approach to Numerical Computing». Em: *SIAM Review* 59.1 (fev. de 2017), pp. 65–98. DOI: 10.1137/141000671.

[2] Gene H Golub e Charles F Van Loan. *Matrix computations*. 4º edição. Johns Hopkins University Press, 2013.

[3] Steven J. Leon. *Algebra linear com aplicações*. Rio de Janeiro: LTC, 2011, p. 464.

[4] Carl D. Meyer. *Matrix analysis and Applied linear algebra*. Philadelphia: SIAM, 2000, p. 869.

[5] David Poole. *Álgebra Linear: uma introdução moderna*. Tradução da 4º edição norte-americana. Cengage Learning, 2017, p. 722.

[6] Taina da Silva. *Resolução de sistemas lineares em $\mathbb{Z}_p$*. Trabalho de conclusão de curso (Licenciatura em Matemática), UFSC (Universidade Federal de Santa Catarina), Blumenau, Brazil. 2022.

[7] David S. Watkins. *Fundamentals of Matrix Computations. Pure and Applied Mathematics: A Wiley Series of Texts, Monographs and Tracts.* Hoboken: Wiley, 2010, p. 644.