Algoritma Penyelesaian Sistem Persamaan Linear *Fuzzy* Dan Implementasinya pada Bahasa C^{++}

Muslim Syahnur[1], Defri Ahmad[2]
[1,2] Program Studi Matematika, Fakultas Matematika Ilmu Pengetahuan dan Alam, Universitas Negeri Padang (UNP)

Article Info

Article history:

Received August 08, 2023
Revised August 29, 2023
Accepted March 20, 2024

Keywords:

Fuzzy Linear System
Crout Decomposition
C^{++} Language

Kata Kunci:

SPL *Fuzzy*
Dekomposisi *Crout*
Bahasa C^{++}

ABSTRACT

A system of fuzzy linear equations is a linear equation with more than one number that is related to one another where the constants and variables are in the form of fuzzy numbers. To solve it, the system of fuzzy linear equations is changed to a system of non-fuzzy linear equations and the solution uses codeblock program software to get fast and accurate results. In solving the system of fuzzy linear equations in this study, the Crout decomposition method was applied and developed an algorithm to be applied to the C++ language. Crout's decomposition method is to decompose a matrix into a multiplication matrix *LU* where the matrix *L* (Lower) is the lower triangular matrix and the matrix *U* (Upper) is the upper triangular matrix whose main diagonal has a value of one. The results of the research are in the form of programs that can find solutions to systems of fuzzy linear equations that have fulfilled the complexity of the function.

ABSTRAK

Sistem persamaan linear *fuzzy* ialah persamaan linear berjumlah lebih dari satu yang saling berkaitan satu sama lainnya dimana konstanta dan variabelnya berupa bilangan *fuzzy*. Untuk menyelesaikannya sistem persamaan linear *fuzzy* dirubah menjadi sistem persamaan linear *non-fuzzy* dan penyelesaiannya menggunakan software program codeblock supaya memperoleh hasil yang cepat dan akurat. Dalam penyelesaian sistem persamaan linear *fuzzy* pada penelitian ini diterapkan metode dekomposisi *Crout* dan menyusun algoritma untuk diterapkan pada bahasa C^{++}. Metode dekomposisi *Crout* yaitu mendekomposisi suatu matriks menjadi perkalian matriks *LU* dimana matriks *L (Lower)* ialah matriks segitiga bawah dan matriks *U (Upper)* ialah matriks segitiga atas yang diagonal utamanya memiliki nilai satu. Hasil dari penelitian berupa program yang dapat menemukan solusi dari sistem persamaan linear *fuzzy* yang sudah memenuhi kompleksitas fungsi.

Penulis pertama/ Corresponding Author:

(Muslim Syahnur)
Program Studi Matematika,Departemen Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam,
Universitas Negeri Padang, Jl. Prof. Dr. Hamka, Air Tawar barat, Padang Utara, Padang, Indonesia. Kode Pos: 25131
Email: muslimsyahnur4@gmail.com

1. PENDAHULUAN

Sistem persamaan linear (SPL) adalah persamaan linear berjumlah lebih dari satu yang saling berkaitan satu sama lainnya untuk menemukan nilai variabel yang memenuhi dari semua persamaan linear tersebut. Suatu SPL terdiri dari sejumlah berhingga variabel dan sejumlah berhingga persamaan linear [1]. Ada berbagai macam koefisien dan konstanta dalam SPL, yaitu ada yang berbentuk bilangan real, bilangan kompleks, dan bilangan *fuzzy* [2]. Dari penjelasan tersebut, SPL *fuzzy* adalah lebih dari satu persamaan linear yang saling berkaitan satu sama lainnya yang konstanta dan variabelnya berupa bilangan *fuzzy*.

Fuzzy diartikan sesuatu yang samar, tidak jelas, kabur, semu dan ketidakpastian [3] sedangkan himpunan *fuzzy* adalah kumpulan objek-objek dengan status keanggotaan yang tidak dapat ditentukan secara tegas [4]. Himpunan *fuzzy* juga didefinisikan sebagai kumpulan dari elemen-elemen yang memiliki tingkatan keanggotaan [5]. Himpunan *fuzzy* dicirikan dengan fungsi keanggotaan yang menegaskan bahwa suatu tingkatan keanggotaan memiliki interval antara 0 sampai 1, sehingga bisa disimpulkan nilai keanggotaan *fuzzy* berada pada interval [0,1] [6].

Terdapat beberapa fungsi keanggotaan yang dapat diterapkan untuk mendapatkan nilai keanggotaan, yaitu representasi linear, representasi segitiga, representasi kurva trapesium, representasi kurva-S dan representasi kurva bentuk lonceng [7]. Fungsi keanggotaan yang diterapkan pada penelitian ini yaitu fungsi keanggotaan bentuk segitiga, karena bentuknya sederhana, memenuhi syarat keanggotaan bilangan *fuzzy* dan mewakili representasi fungsi keanggotaan bentuk yang lainnya [8].

Penyelesaian SPL *fuzzy* dapat dilakukan menggunakan dua metode yaitu metode langsung dan tidak langsung. Metode langsung yang biasanya disebut metode eksak, terdiri dari metode invers, eliminasi, subtitusi, dekomposisi *LU*, dekomposisi Cholesky, dekomposisi *QR*, dekomposisi *Crout* dan dekomposisi *ST*. Metode tidak langsung disebut metode iterasi, terdiri dari metode iterasi Jacobi, metode Newton dan metode Gauss Seidel [9]. Metode yang diterapkan pada penelitian ini ialah metode dekomposisi *Crout* karena metode dekomposisi *Crout* memiliki ketepatan dalam memperoleh solusi dari SPL dibuktikan dengan operasi aritmatika metode dekomposisi *Crout* yang membutuhkan tingkat ketelitian lebih dikarenakan proses perhitungannya lebih panjang [10]. Selain itu, berdasarkan analisis algoritma dan waktu eksekusi, metode dekomposisi *Crout* lebih unggul dari metode yang lainnya karena dapat menyelesaikan SPL berukuran besar [11].

Beberapa penelitian yang membahas mengenai penyelesaian SPL *fuzzy* sudah ada sebelumnya, yaitu Permata (2018) [3] menyatakan bahwa SPL *fuzzy* dapat diselesaikan menggunakan metode Dekomposisi Crout dan menurut Irmawati (2017) [12] solusi SPL *fuzzy* dapat dicari menggunakan pemrograman matlab. Pada penelitian ini dikaji algoritma penyelesaian SPL *fuzzy* menggunakan metode Dekomposisi Crout, kemudian algoritma tersebut akan dikonversikan menggunakan bahasa C^{++}.

Algoritma merupakan alur pemikiran atau susunan langkah dalam menyelesaikan suatu masalah secara sistematika dan logis [13]. Dikatakan algoritma karena alur pemikiran penyelesaian masalah tersebut ditulis dalam bentuk yang sistematis sehingga mudah untuk diterapkan menjadi sebuah program menggunakan bahasa pemrograman [14]. Penyelesaian SPL *fuzzy* menggunakan metode dekomposisi *Crout* diawali dengan menentukan *n* persamaan dan *n* variabel dari SPL *fuzzy* kemudian mengubah persamaannya ke matriks $\boldsymbol{A\widetilde{X}} = \boldsymbol{\widetilde{Y}}$, dilanjutkan dengan mengubah bentuk dari SPL *fuzzy* ke SPL *fuzzy* fungsi keanggotaan segitiga dengan potongan-α dan mengubah SPL *fuzzy* ke SPL *non-fuzzy* dengan mengubah SPL *fuzzy n* persamaan dan *n* variabel ke *2n* persamaan dan *2n* variabel, memfaktorkan matriks *S* tersebut sehingga matriks S menjadi $S = LU$ dengan menggunakan metode dekomposisi *Crout* dimana matriks U merupakan matriks segitiga atas yang diagonal utamanya bernilai satu, dan diakhiri dengan menentukan $\boldsymbol{\widetilde{X}}$ dari operasi matriks $\boldsymbol{LU\widetilde{X}} = \boldsymbol{\widetilde{Y}}$ menggunakan teknik penyulihan maju dan penulihan mundur. Untuk mempermudah perhitungan penyelesaian SPL *fuzzy* tersebut maka dibuat program dengan bahasa C^{++}.

Bahasa C^{++} digunakan dalam penelitian ini karena bahasa C^{++} adalah bahasa pemrograman komputer yang sudah mempunyai bahasa tingkat tinggi *(High Level Language)* [15], artinya manusia mudah untuk memahami dan mempelajari bahasa pemrograman ini. Menggunakan bahasa C^{++}, secara tidak langung telah menguasai dasar logika dari bahasa pemrograman karena secara garis besar, bahasa pemrograman lainnya memiliki kemiripan dengan bahasa C^{++} [16]. Selain itu, bahasa C^{++} dapat meningkatkan produktivitas pemrograman dibandingkan bahasa prosedural seperti bahasa C, PASCAL dan BASIC karena kode dapat digunakan kembali [17].

2. METODE

Jenis penelitian ini adalah penelitian teoritis dasar. Pada penelitian ini terdapat langkah-langkah yang dilakukan yaitu: memahami konsep SPL *fuzzy*. Kemudian menyusun algoritma penyelesaian SPL *fuzzy* menggunakan metode dekomposisi *Crout*. Metode dekomposisi *Crout* adalah mendekomposisi suatu matriks menjadi perkaliak matriks *LU* dimana matriks *L (Lower)* ialah matriks segitiga bawah dan matriks *U* (*Upper)* ialah matriks segitiga atas yang diagonal utamanya memiliki nilai satu. Selanjutnya merancang *pseudocode* penyelesaian SPL *fuzzy* sebagai bagan alur yang berperan untuk menerjemahkan proses berjalannya program supaya lebih mudah dipahami. Kemudian membuat program penyelesaian SPL *fuzzy* menggunakan bahasa pemrograman C^{++}. Selanjutnya menguji program penyelesaian SPL *fuzzy* apakah bisa menyelesaikan soal yang diberikan yaitu soal SPL *fuzzy* jika belum maka melakukan perbaikan agar program bisa menyelesaikan soal yang diberikan.

3. HASIL DAN PAMBAHASAN

3.1 Menyelesaikan Sistem Persamaan Linear Fuzzy

Langkah-langkah dalam menyelesaikan SPL *fuzzy* metode dekomposisi *Crout* yaitu:
Menentukan *n* persamaan dan *n* variabel dari SPL *fuzzy* kemudian mengubah persamaannya ke matriks $\boldsymbol{A\widetilde{X}} = \boldsymbol{\widetilde{Y}}$
Bentuk umum SPL *fuzzy* yaitu:

$$\begin{aligned} a_{11}\tilde{x}_1 + a_{12}\tilde{x}_2 + \cdots + a_{1n}\tilde{x}_n &= \tilde{y}_1 \\ a_{21}\tilde{x}_1 + a_{22}\tilde{x}_2 + \cdots + a_{2n}\tilde{x}_n &= \tilde{y}_2 \\ &\vdots \\ a_{n1}\tilde{x}_n + a_{n2}\tilde{x}_2 + \cdots + a_{nn}\tilde{x}_n &= \tilde{y}_n. \end{aligned} \tag{1}$$

Dimana $\tilde{x}_1, \tilde{x}_2, \ldots, \tilde{x}_n$ merupakan variabel-variabel dari bilangan *fuzzy* dan $\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_n$merupakan konstanta dari bilangan *fuzzy* dan a_{ij} dimana $1 \le i \le n, 1 \le j \le n$ merupakan koefisien dari bilangan *fuzzy* yang merupakan bilangan real. Nyatakan SPL *fuzzy* ke dalam bentuk $\boldsymbol{A\widetilde{X}} = \boldsymbol{\widetilde{Y}}$ yaitu:

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} \tilde{x}_1 \\ \tilde{x}_2 \\ \vdots \\ \tilde{x}_n \end{bmatrix} = \begin{bmatrix} \tilde{y}_1 \\ \tilde{y}_2 \\ \vdots \\ \tilde{y}_n \end{bmatrix}. \tag{2}$$

Mengubah bentuk dari SPL *fuzzy* ke SPL *fuzzy* fungsi keanggotaan segitiga dengan potongan-α. Adapun bentuk dari SPL *fuzzy* fungsi keanggotaan segitiga dengan potongan-α yaitu:

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix} \begin{bmatrix} \left(\underline{x}_1(\alpha), \overline{x}_1(\alpha)\right) \\ \left(\underline{x}_2(\alpha), \overline{x}_2(\alpha)\right) \\ \vdots \\ \left(\underline{x}_n(\alpha), \overline{x}_n(\alpha)\right) \end{bmatrix} = \begin{bmatrix} \left(\underline{y}_1(\alpha), \overline{y}_1(\alpha)\right) \\ \left(\underline{y}_2(\alpha), \overline{y}_2(\alpha)\right) \\ \vdots \\ \left(\underline{y}_n(\alpha), \overline{y}_n(\alpha)\right) \end{bmatrix}. \tag{3}$$

Mengubah SPL *fuzzy* ke SPL *non-fuzzy* yaitu dengan mengubah SPL *fuzzy* *n* persamaan dan *n* variabel ke $2n$ persamaan dan $2n$ variable

$$\boldsymbol{SX^*} = \boldsymbol{Y^*} \tag{4}$$

menjadi

$$\begin{bmatrix} s_{1,1} & \cdots & s_{1,n} & s_{1,n+1} & \cdots & s_{1,2n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ s_{n,1} & \cdots & s_{n,n} & s_{n,n+1} & \cdots & s_{n,2n} \\ s_{n+1,1} & \cdots & s_{n+1,n} & s_{n+1,n+1} & \cdots & s_{n+1,2n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ s_{2n,1} & \cdots & s_{2n,n} & s_{2n,n+1} & \cdots & s_{2n,2n} \end{bmatrix} \begin{bmatrix} \underline{x}_1(\alpha) \\ \vdots \\ \underline{x}_n(\alpha) \\ -\overline{x}_1(\alpha) \\ \vdots \\ -\overline{x}_n(\alpha) \end{bmatrix} = \begin{bmatrix} \underline{y}_1(\alpha) \\ \vdots \\ \underline{y}_n(\alpha) \\ -\overline{y}_1(\alpha) \\ \vdots \\ -\overline{y}_n(\alpha) \end{bmatrix}. \tag{5}$$

Persamaan (5) bukan merupakan SPL *fuzzy*. Elemen-elemen matriks S memiliki syarat yaitu:
1) Apabila $a_{i,j} \geq 0$ maka $s_{i,j} = s_{n+i,n+j} = a_{i,j}$
2) Apabila $a_{i,j} < 0$ maka $s_{i,n+j} = s_{n+i,j} = -a_{i,j}$
3) Elemen lainnya bernilai nol
Berdasarkan syarat dari elemen-elemen matriks S tersebut maka:

$$S_1 = \begin{bmatrix} s_{1,1} & \cdots & s_{1,n} \\ \vdots & \ddots & \vdots \\ s_{n,1} & \cdots & s_{n,n} \end{bmatrix} = \begin{bmatrix} s_{n+1,n+1} & \cdots & s_{n+1,2n} \\ \vdots & \ddots & \vdots \\ s_{2n,n+1} & \cdots & s_{2n,2n} \end{bmatrix} \tag{6}$$

dan untuk

$$S_2 = \begin{bmatrix} s_{1,n+1} & \cdots & s_{1,2n} \\ \vdots & \ddots & \vdots \\ s_{n,n+1} & \cdots & s_{n,2n} \end{bmatrix} = \begin{bmatrix} s_{n+1,1} & \cdots & s_{n+1,n} \\ \vdots & \ddots & \vdots \\ s_{2n,1} & \cdots & s_{2n,n} \end{bmatrix} \tag{7}$$

sehingga diperoleh

$$S = \begin{bmatrix} S_1 & S_2 \\ S_2 & S_1 \end{bmatrix}. \tag{8}$$

Maka dari itu sistem persamaan tersebut dapat ditulis sebagai berikut:

$$\begin{bmatrix} S_1 & S_2 \\ S_2 & S_1 \end{bmatrix} \begin{bmatrix} \underline{X} \\ \overline{X} \end{bmatrix} = \begin{bmatrix} \underline{Y} \\ \overline{Y} \end{bmatrix}. \tag{9}$$

Selanjutnya memfaktorkan matriks S menjadi matriks $S = LU$. Menggunakan metode dekomposisi *Crout* untuk mendapatkan matriks $S = LU$ dari pemfaktoran matriks S sehingga diperoleh SPL $\boldsymbol{LU\widetilde{X} = \widetilde{Y}}$.

$$S = LU \tag{10}$$

$$\begin{bmatrix} s_{1,1} & \cdots & s_{1,n} & s_{1,n+1} & \cdots & s_{1,2n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ s_{n+1,1} & \cdots & s_{n,n} & s_{n,n+1} & \cdots & s_{n,2n} \\ s_{n+1,1} & \cdots & s_{n+1,n} & s_{n+1,n+1} & \cdots & s_{n+1,2n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ s_{2n,1} & \cdots & s_{2n,n} & s_{2n,n+1} & \cdots & s_{2n,2n} \end{bmatrix} =$$

$$\begin{bmatrix} l_{1,1} & 0 & 0 & 0 & 0 & 0 \\ \vdots & \ddots & 0 & 0 & 0 & 0 \\ l_{n,1} & \cdots & l_{n,n} & 0 & 0 & 0 \\ l_{n+1,1} & \cdots & l_{n+1,n} & l_{n+1,n+1} & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & 0 \\ l_{2n,1} & \cdots & l_{2n,n} & l_{2n,n+1} & \cdots & l_{2n,2n} \end{bmatrix} \begin{bmatrix} u_{1,1} & u_{1,2} & \cdots & u_{1,n+1} & \cdots & u_{1,2n} \\ 0 & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & u_{n,n} & u_{n,n+1} & \cdots & u_{n,2n} \\ 0 & 0 & 0 & u_{n+1,n+1} & \cdots & u_{n+1,2n} \\ 0 & 0 & 0 & 0 & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & u_{2n,2n} \end{bmatrix} \quad (11)$$

dengan

$$S = \begin{bmatrix} s_{1,1} & \cdots & s_{1,n} & s_{1,n+1} & \cdots & s_{1,2n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ s_{n+1,1} & \cdots & s_{n,n} & s_{n,n+1} & \cdots & s_{n,2n} \\ s_{n+1,1} & \cdots & s_{n+1,n} & s_{n+1,n+1} & \cdots & s_{n+1,2n} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ s_{2n,1} & \cdots & s_{2n,n} & s_{2n,n+1} & \cdots & s_{2n,2n} \end{bmatrix}, \quad (12)$$

$$L = \begin{bmatrix} l_{1,1} & 0 & 0 & 0 & 0 & 0 \\ \vdots & \ddots & 0 & 0 & 0 & 0 \\ l_{n,1} & \cdots & l_{n,n} & 0 & 0 & 0 \\ l_{n+1,1} & \cdots & l_{n+1,n} & l_{n+1,n+1} & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & 0 \\ l_{2n,1} & \cdots & l_{2n,n} & l_{2n,n+1} & \cdots & l_{2n,2n} \end{bmatrix}, \quad (13)$$

dan

$$U = \begin{bmatrix} u_{1,1} & u_{1,2} & \cdots & u_{1,n+1} & \cdots & u_{1,2n} \\ 0 & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & u_{n,n} & u_{n,n+1} & \cdots & u_{n,2n} \\ 0 & 0 & 0 & u_{n+1,n+1} & \cdots & u_{n+1,2n} \\ 0 & 0 & 0 & 0 & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & u_{2n,2n} \end{bmatrix}. \quad (14)$$

dalam menyelesaikan SPL ini menggunakan metode dekomposisi *Crout* maka $u_{i,j} = 1$ untuk setiap $i = j$ dimana $1 \leq i \leq 2n$ dan $1 \leq j \leq 2n$.
Selanjutnya menentukan $\widetilde{\boldsymbol{X}}$ dari operasi matriks $\boldsymbol{LU}\widetilde{\boldsymbol{X}} = \widetilde{\boldsymbol{Y}}$. Dengan memisalkan $\boldsymbol{U}\widetilde{\boldsymbol{X}} = \widetilde{\boldsymbol{B}}$ sehingga $\boldsymbol{L}\widetilde{\boldsymbol{B}} = \widetilde{\boldsymbol{Y}}$ dengan teknik penyulihan maju dan $\boldsymbol{U}\widetilde{\boldsymbol{X}} = \widetilde{\boldsymbol{B}}$ dengan teknik penyulihan mundur. Sehingga sistem persamaan yang terbentuk adalah seperti berikut:

$$S = \boldsymbol{LU}\widetilde{\boldsymbol{X}} = \widetilde{\boldsymbol{Y}}. \quad (15)$$

Misalkan $\boldsymbol{U}\widetilde{\boldsymbol{X}} = \widetilde{\boldsymbol{B}}$ maka

$$\boldsymbol{L}\widetilde{\boldsymbol{B}} = \widetilde{\boldsymbol{Y}}. \quad (16)$$

Selanjutnya, menentukan nilai dari vektor kolom $\widetilde{\boldsymbol{B}}$. Untuk menentukannya matriks L, matriks $\widetilde{\boldsymbol{B}}$ dan matriks $\widetilde{\boldsymbol{Y}}$ perlu dipartisi supaya mudah dalam mencari nilainya, maka

$$\boldsymbol{L}\widetilde{\boldsymbol{B}} = \widetilde{\boldsymbol{Y}} \quad (17)$$

$$\begin{bmatrix} L_1 & 0 \\ L_2 & L_3 \end{bmatrix} \begin{bmatrix} \underline{B} \\ \overline{B} \end{bmatrix} = \begin{bmatrix} \underline{Y} \\ \overline{Y} \end{bmatrix} \quad (18)$$

$$\begin{bmatrix} L_1.\underline{B} \\ L_2.\underline{B} + L_3.\overline{B} \end{bmatrix} = \begin{bmatrix} \underline{Y} \\ \overline{Y} \end{bmatrix} \quad (19)$$

dengan

$$L = \begin{bmatrix} L_1 & 0 \\ L_2 & L_3 \end{bmatrix} = \begin{bmatrix} l_{1,1} & 0 & 0 & 0 & 0 & 0 \\ \vdots & \ddots & 0 & 0 & 0 & 0 \\ l_{n,1} & \cdots & l_{n,n} & 0 & 0 & 0 \\ l_{n+1,1} & \cdots & l_{n+1,n} & l_{n+1,n+1} & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & 0 \\ l_{2n,1} & \cdots & l_{2n,n} & l_{2n,n+1} & \cdots & l_{2n,2n} \end{bmatrix} \quad (20)$$

$$\tilde{B} = \begin{bmatrix} \underline{B} \\ \overline{B} \end{bmatrix} = \begin{bmatrix} \underline{b}_1(\alpha) \\ \vdots \\ \underline{b}_n(\alpha) \\ \overline{b}_1(\alpha) \\ \vdots \\ \overline{b}_n(\alpha) \end{bmatrix} \quad (21)$$

dan

$$\tilde{Y} = \begin{bmatrix} \underline{Y} \\ \overline{Y} \end{bmatrix} = \begin{bmatrix} \underline{y}_1(\alpha) \\ \vdots \\ \underline{y}_n(\alpha) \\ -\overline{y}_1(\alpha) \\ \vdots \\ -\overline{y}_n(\alpha) \end{bmatrix}. \quad (22)$$

Sehingga didapatkan beberapa persamaan seperi berikut:

$$L_1 \cdot \underline{B} = \underline{Y} \quad (23)$$

$$\begin{bmatrix} l_{1,1} & 0 & 0 \\ \vdots & \ddots & 0 \\ l_{n,1} & \cdots & l_{n,n} \end{bmatrix} \begin{bmatrix} \underline{b}_1(\alpha) \\ \vdots \\ \underline{b}_n(\acute{a}) \end{bmatrix} = \begin{bmatrix} \underline{y}_1(\alpha) \\ \vdots \\ \underline{y}_n(\alpha) \end{bmatrix} \quad (24)$$

dan

$$L_2 \cdot \underline{B} + L_3 \cdot \overline{B} = \overline{Y} \quad (25)$$

$$\begin{bmatrix} l_{n+1,1} & \cdots & l_{n+1,n} \\ \vdots & \ddots & \vdots \\ l_{2n,1} & \cdots & l_{2n,n} \end{bmatrix} \begin{bmatrix} \underline{b}_1(\alpha) \\ \vdots \\ \underline{b}_n(\alpha) \end{bmatrix} + \begin{bmatrix} l_{n+1,n+1} & 0 & 0 \\ \vdots & \ddots & 0 \\ l_{2n,n+1} & \cdots & l_{2n,2n} \end{bmatrix} \begin{bmatrix} \overline{b}_1(\alpha) \\ \vdots \\ \overline{b}_n(\alpha) \end{bmatrix} = \begin{bmatrix} \overline{y}_1(\alpha) \\ \vdots \\ \overline{y}_n(\alpha) \end{bmatrix}. \tag{26}$$

Langkah berikutnya adalah menetapkan nilai matriks $\widetilde{\boldsymbol{X}}$ atau solusi dari SPL. Ditemukan bahwa:

$$\boldsymbol{U}\widetilde{\boldsymbol{X}} = \widetilde{\boldsymbol{B}} \tag{27}$$

lalu dipartisi sehingga

$$\begin{bmatrix} U_1 & U_2 \\ 0 & U_3 \end{bmatrix} \begin{bmatrix} \underline{X} \\ \overline{X} \end{bmatrix} = \begin{bmatrix} \underline{B} \\ \overline{B} \end{bmatrix} \tag{28}$$

$$\begin{bmatrix} U_1.\underline{X} + U_2.\overline{X} \\ U_3.\overline{X} \end{bmatrix} = \begin{bmatrix} \underline{B} \\ \overline{B} \end{bmatrix} \tag{29}$$

maka diperoleh

$$U_3.\overline{X} = \overline{B} \tag{30}$$

$$\begin{bmatrix} u_{n+1,n+1} & \cdots & u_{n+1,2n} \\ 0 & \ddots & \vdots \\ 0 & 0 & u_{2n,2n} \end{bmatrix} \begin{bmatrix} \overline{x}_1(\alpha) \\ \vdots \\ \overline{x}_n(\alpha) \end{bmatrix} = \begin{bmatrix} \overline{b}_1(\alpha) \\ \vdots \\ \overline{b}_n(\alpha) \end{bmatrix} \tag{31}$$

dengan

$$u_{n+1,n+1}, u_{n+2,n+2}, \cdots, u_{2n,2n} = 1 \tag{32}$$

dan

$$U_1.\underline{X} + U_2.\overline{X} = \underline{B} \tag{33}$$

$$\begin{bmatrix} u_{1,1} & u_{1,2} & u_{1,n} \\ 0 & \ddots & \vdots \\ 0 & 0 & u_{n,n} \end{bmatrix} \begin{bmatrix} \underline{x}_1(\alpha) \\ \vdots \\ \underline{x}_n(\alpha) \end{bmatrix} + \begin{bmatrix} u_{1,n+1} & \cdots & u_{1,2n} \\ \vdots & \ddots & \vdots \\ u_{n,n+1} & \cdots & u_{n,2n} \end{bmatrix} \begin{bmatrix} \overline{x}_1(\alpha) \\ \vdots \\ \overline{x}_n(\alpha) \end{bmatrix} = \begin{bmatrix} \underline{b}_1(\alpha) \\ \vdots \\ \underline{b}_n(\alpha) \end{bmatrix} \tag{34}$$

dengan

$$u_{1,1}, u_{2,2}, \cdots, u_{n,n} = 1. \tag{35}$$

Contoh:
Menyelesaikan SPL *fuzzy* menggunakan metode dekomposisi *Crout* sebagai berikut: Diberikan SPL *fuzzy* yang mana pada kali ini dengan dua persamaan dan dua variabel

$$\begin{aligned} \tilde{x}_1 + 2\tilde{x}_2 &= \tilde{8} \\ 2x_1 - \tilde{x}_2 &= \tilde{4} \end{aligned} \tag{36}$$

dengan bilangan *fuzzy* $\tilde{8}$ dan $\tilde{4}$ yang memiliki fungsi keanggotaan segitiga sebagai berikut:

$\tilde{8}$ = segitiga (x; 6, 8, 10)
$\tilde{4}$ =segitiga (x; 2,4,6)
Bagaimana solusi dari SPL berikut dengan menggukana metode dekomposisi *Crout*?
Penyelesaian:
Pertama merubah bentuk variabel dan konstanta yang merupakan bilangan *fuzzy* dari SPL ke dalam bentuk potongan-α, sebagai berikut:

a) Fungsi keanggotaan segitiga bilangan *fuzzy* $\tilde{8}$:

$$\tilde{\mu}_8 = \text{segitiga } (x; 6, 8, 10) = \begin{cases} \frac{x-6}{2}, & \text{jika } 6 \leq x \leq 8 \\ \frac{10-x}{2}, & \text{jika } 8 \leq x \leq 10 \\ 0, & \text{jika } x \leq 6 \text{ dan } x \geq 10 \end{cases} \quad (37)$$

Sehingga didapatkan α-cut dari bilangan *fuzzy* $\tilde{8}$ untuk $\alpha \in [0, 1]$ yaitu dengan mengumpakaman: $\alpha = \frac{x-6}{2}$ dan $\alpha = \frac{10-x}{2}$ maka diperoleh $x = 2\alpha + 6$ dan $x = 10 - 2\alpha$ sehingga didapatkan $8_\alpha = [2\alpha + 6, 10 - 2\alpha]$.

b) Fungsi keanggotaan segitiga bilangan *fuzzy* $\tilde{4}$:

$$\tilde{\mu}_4 = \text{segitiga } (x; 2, 4, 6) = \begin{cases} \frac{x-2}{2}, & \text{jika } 2 \leq x \leq 4 \\ \frac{6-x}{2}, & \text{jika } 4 \leq x \leq 6 \\ 0, & \text{jika } x \leq 2 \text{ dan } x \geq 6 \end{cases} \quad (38)$$

Sehingga didapatkan α-cut dari bilangan *fuzzy* $\tilde{4}$ untuk $\alpha \in [0,1]$ yaitu dengan mengumpamakan: $\alpha = \frac{x-2}{2}$ dan $\alpha = \frac{6-x}{2}$ maka diperoleh $x = 2\alpha + 2$ dan $x = 6 - 2\alpha$ sehingga didapatkan $4_\alpha = [2\alpha + 2, 6 - 2\alpha]$.

Dari (37) dan (38) maka SPL *fuzzy* menjadi

$$\begin{aligned} &\left(\underline{x}_1(\alpha), \overline{x}_1(\alpha)\right) + 2\left(\underline{x}_2(\alpha), \overline{x}_2(\alpha)\right) = [2\alpha + 6, 10 - 2\alpha] \\ &2\left(\underline{x}_1(\alpha), \overline{x}_1(\alpha)\right) - \left(\underline{x}_2(\alpha), \overline{x}_2(\alpha)\right) = [2\alpha + 2, 6 - 2\alpha]. \end{aligned} \quad (39)$$

Selanjutnya mengganti sistem persamaan ke dalam bentuk persamaan matriks $\boldsymbol{A\tilde{X}} = \boldsymbol{\tilde{Y}}$ maka

$$\begin{bmatrix} 1 & 2 \\ 2 & -1 \end{bmatrix} \begin{bmatrix} \underline{x}_1(\alpha), \overline{x}_1(\alpha) \\ \underline{x}_2(\alpha), \overline{x}_2(\alpha) \end{bmatrix} = \begin{bmatrix} 2\alpha + 6, 10 - 2\alpha \\ 2\alpha + 2, 6 - 2\alpha \end{bmatrix}. \quad (40)$$

Selanjutnya mengganti matriks *A* yang memiliki ordo 2×2 menjadi matriks *S* yang memiliki ordo 4×4 dengan syarat elemennya terdapat pada persamaan sebelumnya maka diperoleh:

$$S = \begin{bmatrix} 1 & 2 & 0 & 0 \\ 2 & 0 & 0 & 1 \\ 0 & 0 & 1 & 2 \\ 0 & 1 & 2 & 0 \end{bmatrix}. \quad (41)$$

Sehingga sistem persamaan (40) menjadi $\boldsymbol{SX}^* = \boldsymbol{Y}^*$

$$\begin{bmatrix} 1 & 2 & 0 & 0 \\ 2 & 0 & 0 & 1 \\ 0 & 0 & 1 & 2 \\ 0 & 1 & 2 & 0 \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \\ -\overline{x}_1 \\ -\overline{x}_2 \end{bmatrix} = \begin{bmatrix} 2\alpha + 6 \\ 2\alpha + 2 \\ 2\alpha - 10 \\ 2\alpha - 6 \end{bmatrix}. \tag{42}$$

Selanjutnya yaitu memfaktorkan matriks *S* menjadi perkalian matriks *LU* dengan menggunakan metode dekomposisi *Crout* sehingga diperoleh:

$$S = LU \tag{43}$$

$$\begin{bmatrix} 1 & 2 & 0 & 0 \\ 2 & 0 & 0 & 1 \\ 0 & 0 & 1 & 2 \\ 0 & 1 & 2 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & -4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 2 & -3{,}75 \end{bmatrix} \begin{bmatrix} 1 & 2 & 0 & 0 \\ 0 & 1 & 0 & -0{,}25 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{44}$$

Sehingga sistem persamaannya menjadi

$$\boldsymbol{LUX}^* = \boldsymbol{Y}^* \tag{45}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & -4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 2 & -3{,}75 \end{bmatrix} \begin{bmatrix} 1 & 2 & 0 & 0 \\ 0 & 1 & 0 & -0{,}25 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \\ -\overline{x}_1 \\ -\overline{x}_2 \end{bmatrix} = \begin{bmatrix} 2\alpha + 6 \\ 2\alpha + 2 \\ 2\alpha - 10 \\ 2\alpha - 6 \end{bmatrix}. \tag{46}$$

Selanjutnya yaitu dengan memisalkan $\boldsymbol{UX}^* = \boldsymbol{B}$ maka

$$\boldsymbol{LB} = \boldsymbol{Y}^* \tag{47}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 2 & -4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 2 & -3{,}75 \end{bmatrix} \begin{bmatrix} \underline{b}_1 \\ \underline{b}_2 \\ \overline{b}_1 \\ \overline{b}_2 \end{bmatrix} = \begin{bmatrix} 2\alpha + 6 \\ 2\alpha + 2 \\ 2\alpha - 10 \\ 2\alpha - 6 \end{bmatrix}. \tag{48}$$

Sehingga diperoleh

$$B = \begin{bmatrix} \underline{b}_1 \\ \underline{b}_2 \\ \overline{b}_1 \\ \overline{b}_2 \end{bmatrix} = \begin{bmatrix} 2\alpha + 6 \\ \frac{1}{2}\alpha + \frac{5}{2} \\ 2\alpha - 10 \\ \frac{2}{3}\alpha - \frac{46}{15} \end{bmatrix}. \tag{49}$$

Selanjutnya menentukan nilai $\widetilde{\boldsymbol{X}}$ dengan menerapkan teknik penyulihan mundur. Ditemukan bahwa $\boldsymbol{U}\widetilde{\boldsymbol{X}} = \widetilde{\boldsymbol{B}}$ sehingga

$$\begin{bmatrix} 1 & 2 & 0 & 0 \\ 0 & 1 & 0 & -0{,}25 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \\ -\overline{x}_1 \\ -\overline{x}_2 \end{bmatrix} = \begin{bmatrix} \underline{b}_1 \\ \underline{b}_2 \\ \overline{b}_1 \\ \overline{b}_2 \end{bmatrix} \tag{50}$$

$$\begin{bmatrix} 1 & 2 & 0 & 0 \\ 0 & 1 & 0 & -0{,}25 \\ 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \\ -\overline{x}_1 \\ -\overline{x}_2 \end{bmatrix} = \begin{bmatrix} 2\alpha + 6 \\ \frac{1}{2}\alpha + \frac{5}{2} \\ 2\alpha - 10 \\ \frac{2}{3}\alpha - \frac{46}{15} \end{bmatrix}. \tag{51}$$

Selanjutnya menyetarakan matriks sehingga diperoleh

$$\begin{aligned} \underline{x}_1 &= \frac{2}{3}\alpha + \frac{38}{15} \\ \underline{x}_2 &= \frac{2}{3}\alpha + \frac{26}{15} \\ -\overline{x}_1 &= \frac{2}{3}\alpha - \frac{58}{15} \\ -\overline{x}_2 &= \frac{2}{3}\alpha - \frac{46}{15}. \end{aligned} \tag{52}$$

Maka dari itu didapatkan solusi

$$\begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \\ -\overline{x}_1 \\ -\overline{x}_2 \end{bmatrix} = \begin{bmatrix} \frac{2}{3}\alpha + \frac{38}{15} \\ \frac{2}{3}\alpha + \frac{26}{15} \\ \frac{2}{3}\alpha - \frac{58}{15} \\ \frac{2}{3}\alpha - \frac{46}{15} \end{bmatrix}. \tag{53}$$

3.2 Hasil Program

Program ini dibuat khusus untuk *n* persamaan dan *n* variabel, untuk mencapai hasil optimal, penelitian ini dibatasi pada beberapa kriteria pokok sebagai berikut: Dimensi dari program hasil kompilasi, program yang dibuat melalui bahasa pemrograman tertentu akan diubah menjadi kode-kode yang bisa dimengerti oleh mesin. Ukuran program, apakah besar atau kecil, dapat memengaruhi efisiensi penggunaan memori dan kualitas program secara keseluruhan. Kecepatan eksekusi program juga menjadi faktor penting, di mana suatu program dianggap baik jika mampu menjalankan perintah-perintah yang diberikan dengan cepat. Berikut diberikan SPL *fuzzy* dengan dua persamaan dan dua variabel:

$$\begin{aligned} \tilde{x}_1 + 2\tilde{x}_2 &= \tilde{8} \\ 2\tilde{x}_1 - \tilde{x}_2 &= \tilde{4} \end{aligned} \tag{54}$$

Dengan $\tilde{8}$ dan $\tilde{4}$ adalah bilangan *fuzzy* yang mempunyai fungsi keanggotaan segitiga berikut ini:

$$\begin{aligned} \tilde{8} &= segitiga(x; 6, 8, 10) \\ \tilde{4} &= segitiga(x; 2, 4, 6) \end{aligned} \tag{55}$$

sehingga didapatkan output yaitu:

$$\begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \\ -\overline{x}_1 \\ -\overline{x}_2 \end{bmatrix} = \begin{bmatrix} \frac{2}{3}\alpha + \frac{38}{15} \\ \frac{2}{3}\alpha + \frac{26}{15} \\ \frac{2}{3}\alpha - \frac{58}{15} \\ \frac{2}{3}\alpha - \frac{46}{15} \end{bmatrix}. \tag{56}$$

Selanjutnya diberikan SPL *fuzzy* dengan tiga persamaan dan tiga variabel seperti berikut:

$$\begin{aligned} &\tilde{x}_1 - \tilde{x}_2 + 3\tilde{x}_3 = \tilde{5} \\ &2\tilde{x}_1 + \tilde{x}_2 - 3\tilde{x}_3 = \tilde{4} \\ &\tilde{x}_1 - 2\tilde{x}_2 + \tilde{x}_3 = \widetilde{17} \end{aligned} \tag{57}$$

Dengan $\tilde{5}, \tilde{4}, \widetilde{17}$ adalah bilangan *fuzzy* yang mempunyai fungsi keanggotaan segitiga berikut ini:

$$\begin{aligned} &\tilde{5} = segitiga(x; 2, 5, 8) \\ &\tilde{4} = segitiga\ (x; 2, 4, 6) \\ &\widetilde{17} = segitiga\ (x; 13, 17, 21) \end{aligned} \tag{58}$$

sehingga didapatkan output yaitu:

$$\begin{bmatrix} \underline{x}_1 \\ \underline{x}_2 \\ \underline{x}_3 \\ -\overline{x}_1 \\ -\overline{x}_2 \\ -\overline{x}_3 \end{bmatrix} = \begin{bmatrix} -\alpha + 4 \\ \frac{11}{5}\alpha - \frac{51}{5} \\ \frac{3}{5}\alpha - \frac{13}{5} \\ -\alpha - 2 \\ \frac{11}{5}\alpha + \frac{29}{5} \\ \frac{3}{5}\alpha + \frac{7}{5} \end{bmatrix}. \tag{59}$$

4. KESIMPULAN

Penyelesaian SPL *fuzzy* menggunakan metode dekomposisi *Crout* diawali dengan menentukan *n* persamaan dan *n* variabel dari SPL *fuzzy* kemudian mengubah persamaannya ke matriks $\boldsymbol{A\widetilde{X}} = \boldsymbol{\widetilde{Y}}$, dilanjutkan dengan mengubah bentuk dari SPL *fuzzy* ke SPL *fuzzy* fungsi keanggotaan segitiga dengan potongan-α dan mengubah SPL *fuzzy* ke SPL non-*fuzzy* dengan mengubah SPL *fuzzy n* persamaan dan *n* variabel ke *2n* persamaan dan *2n* variabel, lalu memfaktorkan matriks *S* tersebut sehingga matriks *S* menjadi $S = LU$ dengan menerapkan metode dekomposisi *Crout* dimana matriks *U* merupakan matriks segitiga atas yang diagonal utamanya bernilai satu, dan diakhiri dengan menentukan $\boldsymbol{\widetilde{X}}$ dari operasi matriks $\boldsymbol{LU\widetilde{X}} = \boldsymbol{\widetilde{Y}}$ menggunakan teknik penyulihan maju dan penyulihan mundur. Untuk mempermudah perhitungan penyelesaian SPL *fuzzy* tersebut maka dibuat program dengan bahasa C^{++}. Dengan menggunakan bahasa C^{++} disusun *pseudocode* dari penyelesaian SPL *fuzzy* menggunakan metode dekomposisi *Crout* diperoleh vektor koefisien dan vektor konstanta dari SPL *fuzzy*. Waktu yang dibutuhkan untuk mengeksekusi suatu program atau fungsi dipengaruhi oleh jumlah data input, semakin banyak data input yang diberikan, semakin lama waktu yang dibutuhkan untuk menyelesaikannya dan semakin besar ukuran datanya.

REFERENSI

[1] Putriatama, R. C., & Yohanes, R. S. (2022, February). Mengenalkan Konsep Sistem Persamaan Linear kepada Siswa Sekolah Dasar (Sebuah Kajian Secara Teoritis). In *PRISMA, Prosiding Seminar Nasional Matematika* (Vol. 5, Pp. 371-378).
[2] Fitri, F. A. (2014). Penyelesaian Sistem Persamaan Linear Kompleks Menggunakan Metode. In *Seminar Nasional Teknologi Informasi Komunikasi Dan Industri*.
[3] Permata, A. R., & Arnelis, A. (2018). Penyelesaian Sistem Persamaan Linear Fuzzy Menggunakan Metode Dekomposisi Crout. *Journal of Mathematics UNP*, 3(2).
[4] Zadeh, L. A. (1965). Fuzzy sets. *Information and control*, *8*(3), 338-353.
[5] Davvaz, B., Mukhlash, I., & Soleha, S. (2021). Himpunan Fuzzy dan Rough Sets. *Limits: Journal of Mathematics and Its Applications*, *18*(1), 79-94.
[6] Marzuki, C. C., & Herawati, H. (2015). Penyelesaian sistem persamaan linear fully fuzzy menggunakan metode iterasi jacobi. *Jurnal Sains Matematika dan Statistika*, *1*(1), 1-7.
[7] Praseptyo, C., & Pujiyanta, A. (2014). *Media Pembelajaran Himpunan Fuzzy Berbasis Multimedia* (Doctoral dissertation, Universitas Ahmad Dahlan).
[8] Sari, E. R., & Alisah, E. (2012). Studi Tentang Persamaan Fuzzy. *CAUCHY: Jurnal Matematika Murni dan Aplikasi*, *2*(2), 55-65.

[9] Murni, A. (2015). Pengembangan Software Pemograman Berbasis Pascal Untuk Mengoptimalkan Perkuliahan Metode Numerik. *Semirata*, *5*(1).
[10] Munawaroh, E. L., & Isnarto, I. (2021). Implementasi metode eliminasi Gauss-Jordan dan dekomposisi Crout dalam memprediksi volume lalu lintas. *Unnes Journal of Mathematics*, 28-42.
[11] Supriyono, S., & Syamsudin, D. (2005). Analisis Kinerja Dekomposisi Crout sebagai Penyelesaian Sistem Persamaan Linier Berukuran Besar. In *Seminar Nasional Aplikasi Teknologi Informasi (SNATI).*
[12] Irmawati, I., Sukarsih, I., & Respitawulan, R. (2017). Solusi Sistem Persamaan Linear Fuzzy. *Matematika: Jurnal Teori dan Terapan Matematika*, *16*(2).
[13] Sitorus, L. (2015). *Algoritma dan pemrograman*. Yogyakarta: Andi.
[14] Setyoningrum, N. R., & Rahimma, P. J. (2022, January). Implementasi Algoritma Regresi Linear Dalam Sistem Prediksi Pendaftar Mahasiswa Baru Sekolah Tinggi Teknologi Indonesia Tanjungpinang. In *Prosiding Seminar Nasional Ilmu Sosial dan Teknologi (SNISTEK)* (No. 4, pp. 13-18).
[15] Kadir, A. (2012). *Buku Pintar C++ untuk Pemula*. Yogyakarta: Media Kom.
[16] Dewi, L. J. E. (2010). Media Pembelajaran Bahasa Pemrograman C++. *Jurnal Pendidikan Teknologi dan Kejuruan*, *7*(1).
[17] Hanief, S & Jepriana, W. (2020). *Konsep Algoritme dan Aplikasinya dalam Bahasa C++*. Yogyakarta: ANDI.