

## INVESTIGASI KOMPUTASIONAL METODE LU, QR, SVD, DAN LMS DALAM PENENTUAN BOBOT CMAC

### COMPUTATIONAL INVESTIGATION OF LU, QR, SVD, AND LMS METHODS FOR DETERMINING CMAC WEIGHTS

Muhamad Iradat Achmad<sup>1\*</sup>

<sup>1</sup>Teknik Informatika, Fakultas Teknik, Universitas Dayanu Ikhsanuddin

\*Korespondensi: [irad4t@gmail.com](mailto:irad4t@gmail.com)

diajukan: 15 Juli 2026,

diterima: 5 Agustus 2026.

#### Abstract

Weight determination is an important stage in the implementation of the Cerebellar Model Articulation Controller (CMAC), since the weights determine the output produced from a given input pattern. Several approaches can be employed to obtain CMAC weights, either through direct solution of a system of equations or through iterative learning. This study aims to conduct a computational investigation of the characteristics of CMAC weight solutions obtained using LU, QR, Singular Value Decomposition (SVD), and Least Mean Square (LMS) methods. The experiments were conducted using a Delphi 7-based demonstration application on a sequential image dataset with a size of  $40 \times 32$  pixels and 20 frames, resulting in 25,600 data points. The CMAC structure used a generalization value of ( $G=5$ ) and a displacement vector of  $([1,2,4])$ , producing 1,529 weights. The LMS method used a learning parameter of ( $\mu=0.02$ ). The solutions were compared based on the norm of the weight-vector difference with respect to the SVD solution and the Mean Squared Error (MSE) of the CMAC output. The results show that the norms of the LU, QR, and LMS solutions relative to SVD were 2302.983421, 980.6251922, and 38.91992, respectively, at epoch 8047. The MSE values for SVD, LU, QR, and LMS were 189.06238163, 189.06238877, 189.06238159, and 193.61979177, respectively. The LMS learning process reduced the MSE from 14,699.23164062 to 193.61979177, corresponding to an approximately 98.68% reduction. The investigation demonstrates that the closeness of a weight vector to the SVD solution does not necessarily correspond directly to CMAC output performance in terms of MSE. At epoch 8047, LMS produced the weight vector closest to SVD, whereas LU, QR, and SVD produced nearly identical MSE values. These findings indicate that vector-norm distance and MSE provide two distinct perspectives for evaluating and comparing CMAC weight solutions.

**Keywords:** CMAC; CMAC weights; LU; QR; SVD; LMS; numerical computation; MSE.

#### Abstrak

Penentuan bobot merupakan salah satu tahap penting dalam implementasi Cerebellar Model Articulation Controller (CMAC), karena bobot menentukan keluaran yang dihasilkan berdasarkan pola masukan yang diberikan. Berbagai pendekatan dapat digunakan untuk memperoleh bobot CMAC, baik melalui penyelesaian sistem persamaan secara langsung maupun melalui pembelajaran iteratif. Penelitian ini bertujuan melakukan investigasi komputasional terhadap karakteristik solusi bobot CMAC yang diperoleh menggunakan metode LU, QR, Singular Value Decomposition (SVD), dan Least Mean Square (LMS). Eksperimen dilakukan menggunakan aplikasi demonstrasi berbasis Delphi 7 pada dataset citra sekuensial berukuran  $40 \times 32$  piksel dengan 20 bingkai sehingga diperoleh 25.600 data. Struktur CMAC menggunakan generalisasi ( $G=5$ ) dan displacement vector  $([1,2,4])$ , menghasilkan 1.529 bobot. Metode LMS menggunakan parameter ( $\mu=0,02$ ). Perbandingan solusi dilakukan berdasarkan norma perbedaan vektor bobot terhadap solusi SVD dan Mean Squared Error (MSE) keluaran CMAC. Hasil menunjukkan bahwa norma perbedaan solusi LU, QR, dan LMS terhadap SVD masing-masing sebesar 2302,983421, 980,6251922, dan 38,91992 pada epoch ke-8047. Sementara itu, MSE SVD, LU, QR, dan LMS masing-masing sebesar 189,06238163, 189,06238877, 189,06238159, dan 193,61979177. Proses pembelajaran LMS menurunkan MSE dari 14.699,23164062 menjadi 193,61979177 atau sekitar 98,68%. Hasil investigasi menunjukkan bahwa kedekatan vektor bobot terhadap solusi SVD tidak selalu berbanding langsung dengan performa keluaran CMAC berdasarkan MSE. Pada epoch ke-8047, LMS menghasilkan vektor bobot yang paling dekat terhadap SVD, sedangkan LU, QR, dan SVD menghasilkan MSE yang hampir identik. Temuan ini menunjukkan bahwa norma vektor dan MSE memberikan dua perspektif evaluasi yang berbeda dalam membandingkan solusi bobot CMAC.

**Kata kunci:** CMAC, CMAC; bobot CMAC; LU; QR; SVD; LMS; komputasi numerik; MSE.

## 1. PENDAHULUAN

Cerebellar Model Articulation Controller (CMAC) merupakan salah satu model komputasi adaptif yang dikembangkan untuk merepresentasikan fungsi melalui mekanisme pemetaan masukan ke lokasi-lokasi memori yang berasosiasi dengan keluaran tertentu. Konsep awal CMAC diperkenalkan oleh Albus sebagai pendekatan untuk menghitung fungsi kendali melalui mekanisme associative memory yang memanfaatkan sejumlah lokasi memori untuk merepresentasikan suatu masukan (Albus, 1971)(Albus, 1975a)(Albus, 1975b)

Salah satu karakteristik penting CMAC adalah penggunaan representasi lokal dan beberapa lokasi memori yang secara tumpang susun berkontribusi terhadap suatu keluaran. Struktur tersebut memungkinkan CMAC merepresentasikan fungsi kontinu menggunakan jumlah lokasi penyimpanan yang lebih terkelola dibandingkan pemetaan langsung seluruh ruang masukan (Ker et al., 1997)(Achmad et al., 2004)(Achmad et al., 2014a)(Achmad et al., 2015).

Dalam proses pembelajaran CMAC, bobot yang terkait dengan lokasi memori diperbarui berdasarkan data pelatihan sehingga keluaran model dapat mendekati fungsi atau target yang diinginkan. Proses penyimpanan atau pelatihan pada CMAC dapat dilakukan secara iteratif dan, pada kondisi tertentu, menunjukkan kecenderungan menuju konvergensi (Achmad et al., 2009)(Achmad et al., 2014c).

Dalam perspektif komputasi numerik, penentuan bobot CMAC dapat diformulasikan sebagai permasalahan yang menghubungkan data masukan, struktur pemetaan CMAC, bobot, dan target keluaran. Secara umum, hubungan tersebut dapat dinyatakan dalam bentuk sistem persamaan linear atau permasalahan pendekatan,

$$Aw \approx d$$

dengan **A** menyatakan matriks pemetaan atau aktivasi CMAC, **w** merupakan vektor bobot yang akan ditentukan, dan **d** merupakan vektor target. Formulasi permasalahan least-squares semacam ini dapat diselesaikan menggunakan berbagai pendekatan aljabar linear, termasuk dekomposisi QR dan Singular Value Decomposition (SVD) (Golub & Van Loan, 1989).

Salah satu pendekatan untuk memperoleh solusi sistem persamaan linear adalah faktorisasi LU, yang memfaktorkan matriks ke dalam komponen matriks segitiga sehingga sistem dapat diselesaikan melalui tahapan substitusi (Achmad, 2005)(Achmad, 2021). Pendekatan lainnya adalah faktorisasi QR, yang merepresentasikan matriks sebagai hasil perkalian matriks ortogonal dan matriks segitiga atas dan banyak digunakan dalam penyelesaian permasalahan least-squares. Sementara itu, SVD melakukan dekomposisi matriks ke dalam komponen singular dan merupakan salah satu pendekatan penting dalam penyelesaian permasalahan least-squares, termasuk pada kondisi matriks yang memiliki karakteristik numerik yang kurang baik (Golub & Van Loan, 1989).

Selain pendekatan penyelesaian langsung melalui metode aljabar linear, bobot CMAC dapat diperoleh melalui pendekatan pembelajaran adaptif. Salah satu metode yang relevan adalah Least Mean Square (LMS), yaitu algoritma pembelajaran adaptif yang melakukan pemutakhiran parameter berdasarkan kesalahan antara keluaran sistem dan sinyal target (Achmad et al., 2014b). Berbeda dengan pendekatan LU, QR, dan SVD yang memperoleh solusi melalui proses penyelesaian sistem secara langsung, LMS menghasilkan solusi melalui serangkaian pemutakhiran iteratif. Karakteristik tersebut menyebabkan solusi LMS dapat diamati sebagai suatu proses perubahan bobot dari kondisi awal menuju kondisi yang menghasilkan kesalahan keluaran yang lebih rendah (Achmad et al., 2014c).

Perbedaan mekanisme tersebut menimbulkan persoalan yang menarik ketika keempat metode digunakan untuk menentukan bobot CMAC pada permasalahan yang sama. Metode yang berbeda dapat menghasilkan vektor bobot dengan nilai elemen yang berbeda, tetapi perbedaan tersebut belum tentu menghasilkan perbedaan performa keluaran yang sebanding. Sebaliknya, suatu proses pembelajaran iteratif dapat menghasilkan vektor bobot yang secara bertahap

mendekati solusi yang diperoleh melalui pendekatan penyelesaian langsung. Permasalahan tersebut menjadi dasar dilakukannya investigasi komputasional dalam penelitian ini.

Penelitian ini menggunakan aplikasi demonstrasi yang dikembangkan dengan bahasa pemrograman Delphi 7 (Borland, 2002) untuk membandingkan solusi bobot CMAC yang diperoleh menggunakan metode LU, QR, SVD, dan LMS. Keempat metode diterapkan pada dataset dan konfigurasi CMAC yang sama sehingga karakteristik solusi dapat dibandingkan pada kondisi eksperimen yang setara.

Dataset yang digunakan berupa citra sekuensial dengan lebar 40 piksel, tinggi 32 piksel, dan 20 bingkai, sehingga jumlah keseluruhan data adalah

$$L = 40 \times 32 \times 20 = 25600.$$

Struktur CMAC menggunakan nilai generalisasi  $G=5$  dan vektor displacement

$$\mathbf{d} = [d_1 \quad d_2 \quad d_3] = [1 \quad 2 \quad 4],$$

yang menghasilkan lima lapis bidang bobot tumpang susun dengan total 1.529 bobot. Pada metode LMS digunakan parameter laju pembelajaran  $\mu = 0,02$ . Parameter-parameter tersebut merupakan konfigurasi eksperimen dan implementasi aplikasi yang menjadi objek penelitian.

Investigasi dilakukan dengan membandingkan solusi vektor bobot yang diperoleh dari keempat metode. Solusi SVD digunakan sebagai referensi untuk mengukur kedekatan numerik solusi LU, QR, dan LMS menggunakan norma perbedaan vektor. Selain itu, performa keluaran setiap solusi dievaluasi menggunakan Mean Squared Error (MSE), yaitu ukuran kesalahan kuadrat rata-rata antara keluaran model dan nilai target.

Penggunaan kedua ukuran tersebut dimaksudkan untuk membedakan antara kedekatan numerik vektor bobot dan performa keluaran CMAC. Norma perbedaan terhadap SVD memberikan informasi mengenai seberapa dekat suatu solusi berada terhadap solusi referensi dalam ruang vektor bobot, sedangkan MSE memberikan informasi mengenai tingkat kesalahan keluaran terhadap target.

Penelitian ini secara khusus bertujuan untuk (1) memperoleh dan membandingkan vektor bobot CMAC menggunakan metode LU, QR, SVD, dan LMS, (2) mengukur perbedaan solusi LU, QR, dan LMS terhadap solusi SVD, serta (3) membandingkan performa keempat solusi berdasarkan nilai MSE.

Kontribusi penelitian ini terletak pada investigasi komputasional terhadap hubungan antara perbedaan vektor bobot dan performa keluaran CMAC ketika bobot ditentukan menggunakan metode LU, QR, SVD, dan LMS. Penggunaan aplikasi demonstrasi memungkinkan keempat pendekatan diuji pada dataset dan struktur CMAC yang sama, sehingga karakteristik solusi dapat diamati dan dibandingkan secara kuantitatif.

Penelitian ini difokuskan pada performa solusi vektor bobot secara keseluruhan. Meskipun aplikasi demonstrasi memiliki kemampuan untuk memonitor dinamika setiap elemen bobot selama proses iterasi LMS, analisis rinci mengenai lintasan dan dinamika setiap elemen bobot tidak menjadi ruang lingkup artikel ini. Aspek tersebut akan menjadi dasar untuk pengembangan penelitian lanjutan mengenai karakteristik konvergensi elemen-elemen vektor bobot LMS terhadap solusi SVD.

## 2. METODE PENELITIAN

### 2.1 Desain Penelitian

Penelitian ini menggunakan pendekatan eksperimental-komputasional untuk menginvestigasi perbedaan solusi bobot Cerebellar Model Articulation Controller (CMAC) yang diperoleh melalui metode LU, QR, Singular Value Decomposition (SVD), dan Least Mean Square (LMS). Eksperimen dilakukan dengan menerapkan keempat metode pada dataset dan struktur

CMAC yang sama sehingga solusi yang dihasilkan dapat dibandingkan pada kondisi komputasi yang setara.

CMAC merupakan model jaringan komputasi berbasis associative memory yang memetakan ruang masukan ke sejumlah lokasi memori dan menggunakan bobot yang terkait dengan lokasi tersebut untuk menghasilkan keluaran. Karakteristik pemetaan lokal dan tumpang susun (overlapping) merupakan bagian penting dari mekanisme CMAC dalam merepresentasikan hubungan masukan-keluaran (Achmad et al., 2014a).

Dalam penelitian ini, permasalahan penentuan bobot CMAC diformulasikan sebagai permasalahan komputasi yang menghubungkan vektor masukan, struktur aktivasi atau pemetaan CMAC, vektor bobot, dan target keluaran. Secara umum hubungan tersebut dapat dinyatakan sebagai

$$Aw \approx d$$

dengan  $A$  menyatakan matriks pemetaan CMAC,  $w$  menyatakan vektor bobot, dan  $d$  menyatakan vektor target.

Berdasarkan formulasi tersebut, solusi bobot ditentukan menggunakan dua kelompok pendekatan. Kelompok pertama terdiri atas metode LU, QR, dan SVD yang digunakan untuk memperoleh solusi secara langsung melalui operasi aljabar linear. Kelompok kedua adalah LMS yang memperoleh solusi melalui proses pembelajaran iteratif dengan melakukan pemutakhiran bobot berdasarkan kesalahan keluaran. Pendekatan LMS secara umum digunakan dalam pembelajaran CMAC dan prosesnya dipengaruhi oleh parameter laju pembelajaran (Achmad et al., 2014b).

Eksperimen dilaksanakan menggunakan aplikasi demonstrasi yang dikembangkan dengan bahasa pemrograman Delphi 7. Aplikasi tersebut dirancang untuk menghitung solusi bobot menggunakan keempat metode serta menyediakan hasil komputasi yang dapat dibandingkan pada dataset dan konfigurasi CMAC yang sama.

Evaluasi penelitian difokuskan pada performa solusi vektor bobot secara keseluruhan. Perbandingan dilakukan menggunakan norma perbedaan vektor terhadap solusi SVD dan nilai MSE yang dihasilkan oleh masing-masing metode. Dinamika individual setiap elemen bobot LMS tidak menjadi fokus analisis kuantitatif dalam penelitian ini dan akan menjadi ruang lingkup penelitian lanjutan.

## 2.2 Dataset dan Pembentukan Data Pelatihan

Dataset penelitian berasal dari citra sekuensial yang direpresentasikan dalam tiga dimensi, yaitu posisi kolom, posisi baris, dan nomor bingkai. Setiap sampel data memiliki koordinat masukan tiga dimensi yang dinyatakan sebagai

$$[m, n, o]$$

dengan  $m$  menunjukkan posisi pada arah lebar citra,  $n$  menunjukkan posisi pada arah tinggi citra, dan  $o$  menunjukkan nomor bingkai.

Dataset memiliki ukuran lebar sebesar 40 piksel, tinggi sebesar 32 piksel, dan terdiri atas 20 bingkai. Dengan demikian, jumlah keseluruhan data yang digunakan adalah

$$L = 40 \times 32 \times 20 = 25600.$$

Setiap koordinat  $[m, n, o]$  memiliki nilai target berupa nilai piksel pada posisi tersebut. Dengan demikian, pasangan data pelatihan dapat direpresentasikan sebagai

$$([m, n, o], d_k)$$

dengan  $d_k$  merupakan nilai piksel pada posisi  $[m, n, o]$ .

Dalam implementasi LMS, vektor input dibentuk dari posisi kolom, baris, dan bingkai, sedangkan nilai piksel digunakan sebagai target keluaran. Dengan demikian, dataset pelatihan

tidak hanya merepresentasikan nilai intensitas piksel, tetapi juga mempertahankan informasi posisi spasial dan temporal dari citra sekuensial. Pembentukan dataset tersebut memungkinkan CMAC mempelajari hubungan antara posisi tiga dimensi dan nilai piksel. Selanjutnya, struktur pemetaan CMAC digunakan untuk menghubungkan setiap vektor input dengan sejumlah bobot yang aktif sesuai dengan parameter generalisasi dan displacement vector yang digunakan.

### 2.3 Konfigurasi Struktur CMAC

Struktur CMAC yang digunakan dalam penelitian ini menggunakan nilai generalisasi  $G=5$ . Nilai tersebut menghasilkan lima lapis bidang bobot tumpang susun yang digunakan dalam proses pemetaan data masukan ke ruang bobot. Vektor displacement yang digunakan adalah

$$d = [d_1 \quad d_2 \quad d_3] = [1 \quad 2 \quad 4],$$

dengan  $d_1 = 1$  menyatakan displacement pada sumbu arah lebar,  $d_2 = 2$  pada sumbu arah tinggi, dan  $d_3 = 4$  pada sumbu arah bingkai.

Berdasarkan konfigurasi tersebut, jumlah bobot pada masing-masing lapis adalah 315 untuk Lapis-1, 315 untuk Lapis-2, 360 di Lapis-3, 315 di Lapis-4, dan 224 untuk Lapis-5, sehingga jumlah total bobot di semua Lapis adalah 1529 bobot. Dengan demikian, vektor bobot keseluruhan dapat dituliskan sebagai

$$w = [w_1 \quad w_2 \quad \dots \quad w_{1529}]^T,$$

Jumlah 1.529 elemen tersebut menjadi dimensi vektor solusi yang dibandingkan pada penelitian ini.

Perhitungan jumlah bobot pada masing-masing lapisan dilakukan berdasarkan konfigurasi dimensi data, nilai generalisasi, dan displacement vector. Hasil perhitungan fungsi tersebut menjadi dasar pembentukan struktur ruang bobot yang digunakan oleh metode LU, QR, SVD, maupun LMS.

### 2.4 Penentuan Bobot Menggunakan LU, QR, dan SVD

Setelah struktur CMAC terbentuk, hubungan antara data pelatihan dan bobot direpresentasikan dalam bentuk sistem persamaan linear. Secara umum sistem tersebut dapat dinyatakan sebagai

$$Aw = d$$

atau dalam kondisi data yang tidak menghasilkan penyelesaian eksak sebagai permasalahan pendekatan

$$Aw \approx d$$

Matriks  $A$  merepresentasikan hubungan antara data input dan bobot CMAC yang aktif, sedangkan  $d$  merupakan vektor target dan  $w$  merupakan vektor bobot yang dicari.

Pada pendekatan LU, penyelesaian dilakukan melalui faktorisasi matriks sehingga sistem dapat diselesaikan menggunakan komponen hasil faktorisasi tersebut (Achmad, M. I., 2021). Pada pendekatan QR, matriks difaktorkan menjadi komponen ortogonal dan matriks segitiga atas. Struktur faktorisasi tersebut digunakan untuk memperoleh solusi sistem persamaan linear. Sementara itu, metode SVD melakukan dekomposisi terhadap matriks sistem menjadi komponen-komponen singular. Dalam penelitian ini, solusi yang diperoleh melalui SVD digunakan sebagai solusi referensi untuk membandingkan kedekatan numerik vektor bobot hasil metode lainnya (Golub & Van Loan, 1989). Dengan demikian, ketiga pendekatan menghasilkan masing-masing vektor

$$w_{LU}, w_{QR}, \text{ dan } w_{SVD}.$$

Ketiga vektor tersebut memiliki dimensi yang sama, yaitu 1.529 elemen, sehingga dapat dibandingkan secara langsung.

## 2.5 Penentuan Bobot Menggunakan LMS

Berbeda dengan LU, QR, dan SVD, LMS menentukan bobot melalui proses pembelajaran iteratif. Pada setiap sampel pelatihan, keluaran CMAC dibandingkan dengan target untuk memperoleh kesalahan, kemudian kesalahan tersebut digunakan untuk melakukan pemutakhiran bobot yang aktif. Dalam penelitian ini, parameter laju pembelajaran LMS ditetapkan sebesar

$$\mu = 0.02$$

Implementasi proses pembelajaran dilakukan secara iteratif dengan dataset pelatihan dibentuk dari pasangan vektor input

$$[m, n, o]$$

dan target

$$d_k,$$

dengan  $d_k$  merupakan nilai piksel pada posisi tersebut.

Secara konseptual, kesalahan pembelajaran dapat dipahami sebagai deviasi nilai keluaran dari target yang dapat dinyatakan sebagai

$$e_k = d_k - y_k,$$

dengan  $d_k$  merupakan target dan  $y_k$  merupakan keluaran CMAC.

Pemutakhiran bobot proses belajar dalam implementasi ini dilakukan menggunakan rumus berikut.

$$w_{t+1} = w_t + \frac{2\mu e_t}{G}.$$

Perintah tersebut merepresentasikan mekanisme pemutakhiran bobot LMS pada elemen bobot yang aktif. Dalam implementasi program, pada setiap iterasi,  $G$  bobot yang aktif pada iterasi ke- $t$  dipicu sebagai bobot aktif untuk diperbarui isinya. Besarnya perubahan bobot sesuai dengan formulasi pembelajaran yang diterapkan (Achmad, M. I., 2014b) (Achmad, M. I., 2014c).

Proses pembelajaran dilakukan secara berulang untuk seluruh data pelatihan. Kinerja solusi LMS dievaluasi berdasarkan perubahan MSE selama iterasi. Pada eksperimen yang dilaporkan dalam penelitian ini, proses diamati hingga epoch ke-8047, yang menghasilkan solusi vektor bobot

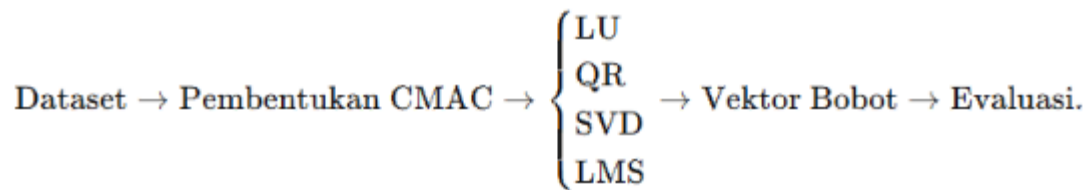
$$w_{LMS..}$$

Parameter dan implementasi LMS tersebut dipertahankan sesuai dengan program demonstrasi yang menjadi objek penelitian.

## 2.6 Prosedur Eksperimen

Prosedur eksperimen yang diperlihatkan dalam diagram alir Gambar 2.1 dilakukan melalui beberapa tahap berikut. Tahap pertama adalah menyiapkan dataset citra sekuensial berukuran  $40 \times 32$  piksel dengan 20 bingkai sehingga diperoleh 25.600 pasangan data input-target. Tahap kedua adalah membentuk struktur CMAC menggunakan nilai generalisasi  $G=5$  dan displacement vector  $[1,2,4]$ . Struktur tersebut menghasilkan lima lapis bidang bobot dengan jumlah keseluruhan 1.529 bobot. Tahap ketiga adalah menentukan solusi bobot menggunakan metode LU, QR, dan SVD. Masing-masing metode menghasilkan vektor bobot dengan dimensi yang sama. Tahap keempat adalah melakukan pembelajaran menggunakan LMS dengan parameter  $\mu = 0.02$ . Proses dilakukan secara iteratif menggunakan pasangan input  $[m,n,o]$  dan target nilai piksel  $d_k$ . Tahap kelima adalah mengevaluasi seluruh solusi yang diperoleh. Solusi SVD digunakan sebagai

referensi untuk menghitung norma perbedaan vektor bobot metode LU, QR, dan LMS. Selanjutnya, MSE digunakan untuk membandingkan performa keluaran keempat solusi.



Gambar 2.1. Diagram alir prosedur eksperimen penelitian

## 2.7 Metrik Evaluasi

Evaluasi solusi dilakukan menggunakan dua ukuran utama, yaitu norma perbedaan vektor bobot terhadap solusi SVD dan Mean Squared Error (MSE).

### 2.7.1 Norma Perbedaan terhadap SVD

Solusi SVD digunakan sebagai referensi untuk mengukur kedekatan numerik solusi lainnya. Norma Euclidean antara vektor bobot suatu metode dan solusi SVD dihitung menggunakan

$$D_X = \|w_X - w_{SVD}\|_2,$$

dengan  $X \in \{LU, QR, LMS\}$ .

Nilai  $D_X$  yang semakin kecil menunjukkan bahwa vektor bobot metode  $X$  semakin dekat secara numerik terhadap solusi SVD. Ukuran ini digunakan untuk mengetahui perbedaan representasi vektor bobot dan tidak dimaksudkan sebagai ukuran langsung kualitas keluaran CMAC.

### 2.7.2 Mean Squared Error (MSE)

Performa keluaran CMAC dievaluasi menggunakan MSE. Secara umum MSE dihitung sebagai

$$MSE = \frac{1}{L} \sum_{k=1}^L (d_k - y_k)^2,$$

dengan  $L=25.600$  merupakan jumlah data,  $d_k$  merupakan target, dan  $y_k$  merupakan keluaran CMAC. Nilai MSE yang lebih kecil menunjukkan tingkat kesalahan kuadrat rata-rata yang lebih rendah antara keluaran CMAC dan target.

Penggunaan kedua metrik tersebut memungkinkan penelitian membedakan dua aspek evaluasi, yaitu seberapa dekat suatu vektor bobot terhadap solusi SVD dan seberapa baik solusi tersebut menghasilkan keluaran CMAC terhadap target.

## 2.8 Ruang Lingkup Analisis

Penelitian ini secara khusus memusatkan analisis pada performa solusi vektor bobot yang dihasilkan oleh metode LU, QR, SVD, dan LMS. Analisis mencakup perbandingan norma vektor terhadap solusi SVD dan perbandingan MSE dari keluaran CMAC.

Aplikasi demonstrasi sebenarnya memiliki kemampuan untuk menampilkan dinamika komputasi selama proses iterasi LMS, termasuk perubahan elemen bobot berdasarkan indeks bobot yang dipilih. Namun, kemampuan tersebut tidak dijadikan fokus analisis utama dalam penelitian ini. Pengamatan terhadap dinamika elemen bobot hanya digunakan sebagai informasi pendukung untuk memahami kecenderungan konvergensi solusi LMS terhadap solusi SVD.

Analisis kuantitatif yang lebih mendalam mengenai dinamika setiap elemen vektor bobot LMS, termasuk lintasan konvergensi masing-masing elemen, perbedaan kecepatan konvergensi,

dan hubungan antara pemutakhiran bobot dengan solusi SVD, ditempatkan sebagai peluang penelitian lanjutan.

### 3. HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil eksperimen dan pembahasan mengenai penentuan bobot *Cerebellar Model Articulation Controller* (CMAC) menggunakan metode LU, QR, SVD, dan LMS. Eksperimen dilakukan pada dataset citra sekuensial dengan konfigurasi dan parameter CMAC yang sama, sehingga perbedaan hasil yang diperoleh dapat diamati berdasarkan karakteristik masing-masing metode penyelesaian.

Fokus utama investigasi diarahkan pada performa solusi vektor bobot yang dihasilkan oleh keempat metode. Performa tersebut dievaluasi melalui perbandingan vektor bobot terhadap solusi SVD sebagai referensi, serta melalui nilai *Mean Squared Error* (MSE) yang dihasilkan oleh masing-masing solusi. Pendekatan ini digunakan untuk mengetahui tidak hanya seberapa besar perbedaan numerik antar-vektor bobot, tetapi juga apakah perbedaan tersebut memberikan pengaruh yang berarti terhadap performa keluaran CMAC.

Eksperimen menggunakan struktur CMAC tiga dimensi yang dibentuk dari data citra sekuensial berukuran  $40 \times 32$  piksel dengan 20 bingkai, sehingga jumlah data yang digunakan sebanyak 25.600 data. Dengan nilai generalisasi ( $G=5$ ) dan vektor *displacement* ( $d=[1,2,4]$ ), struktur CMAC menghasilkan lima lapis bidang bobot dengan total 1.529 bobot. Konfigurasi tersebut dipertahankan sama pada proses penentuan bobot menggunakan metode LU, QR, SVD, maupun LMS untuk memastikan bahwa perbandingan dilakukan pada permasalahan komputasi yang identik.

Hasil metode LU, QR, dan SVD selanjutnya dibandingkan berdasarkan norma perbedaan vektor bobot, sedangkan solusi LMS dievaluasi berdasarkan perubahan MSE dari kondisi awal hingga proses pembelajaran mencapai epoch pengamatan. Perbandingan tersebut menjadi dasar untuk mengidentifikasi kedekatan solusi, tingkat kesalahan, serta karakteristik performa masing-masing metode dalam menentukan bobot CMAC.

Penyajian hasil dilakukan secara bertahap, dimulai dari konfigurasi eksperimen dan struktur CMAC, dilanjutkan dengan hasil penentuan vektor bobot, perbandingan solusi terhadap SVD, perbandingan MSE, evaluasi solusi LMS, dan perbandingan keseluruhan metode. Berdasarkan hasil tersebut, pembahasan diarahkan untuk memperoleh interpretasi komputasional mengenai hubungan antara perbedaan vektor bobot dan performa keluaran CMAC.

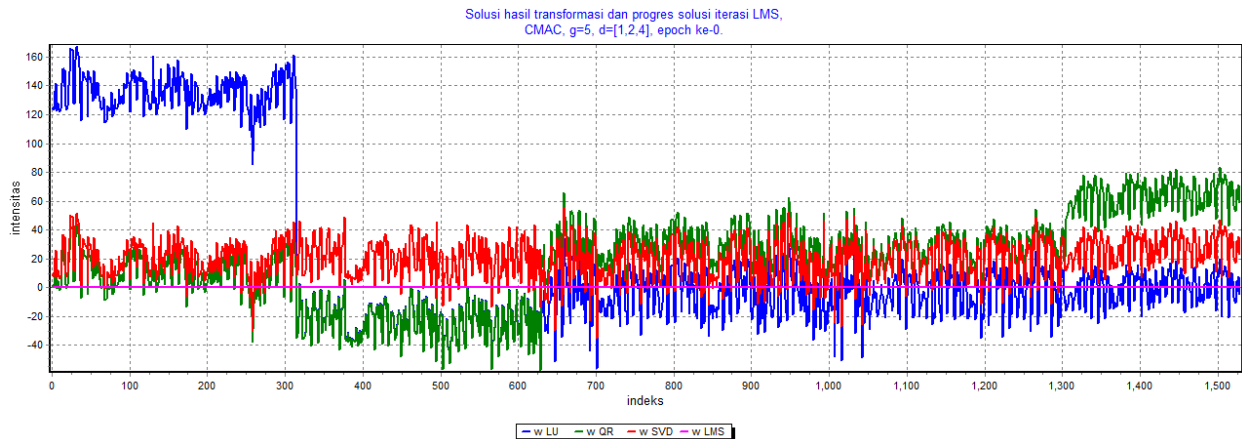
#### 3.1 Hasil Penentuan Vektor Bobot LU, QR, SVD, dan LMS

Berdasarkan skenario eksperimen yang telah ditetapkan, proses penentuan bobot CMAC dilakukan menggunakan empat pendekatan, yaitu LU, QR, SVD, dan LMS. Keempat metode diterapkan pada dataset dan struktur CMAC yang sama, yang terdiri atas 25.600 data pelatihan dan 1.529 parameter bobot. Hasil dari proses tersebut berupa empat vektor bobot yang masing-masing merepresentasikan solusi CMAC berdasarkan karakteristik komputasi metode yang digunakan.

Pada metode LU, QR, dan SVD, vektor bobot diperoleh melalui penyelesaian permasalahan yang dibentuk dari hubungan antara data masukan, fungsi basis CMAC, dan nilai target. Masing-masing metode menggunakan mekanisme komputasi yang berbeda untuk memperoleh solusi. Hasilnya dinyatakan sebagai vektor  $w_{LU}$ ,  $w_{QR}$ , dan  $w_{SVD}$ . Ketiga vektor tersebut memiliki jumlah elemen yang sama, yaitu 1.529 elemen, sesuai dengan jumlah bobot yang dibentuk oleh struktur CMAC, seperti diperlihatkan dalam Gambar 3.1.

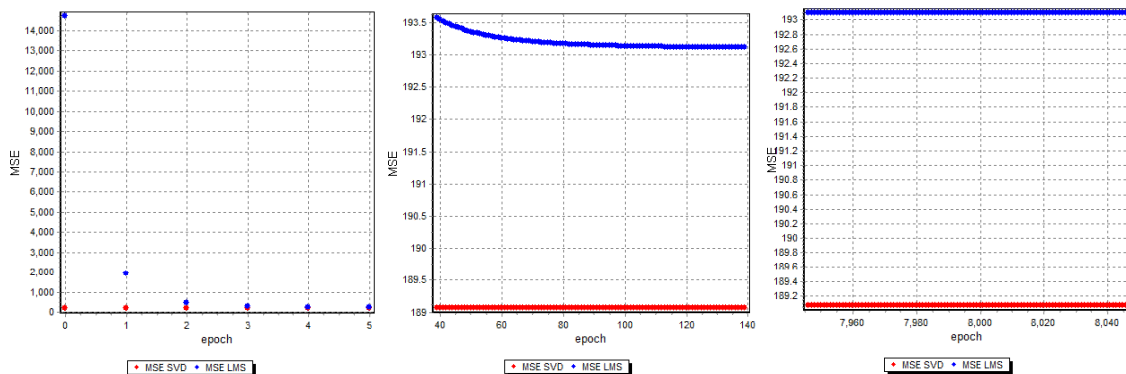
Hasil komputasi menunjukkan bahwa ketiga metode menghasilkan solusi vektor bobot yang tidak harus memiliki nilai elemen yang identik. Perbedaan tersebut selanjutnya dianalisis secara kuantitatif dengan membandingkan masing-masing vektor terhadap solusi SVD. Analisis

tersebut diperlukan karena kesamaan jumlah parameter tidak secara langsung menunjukkan kesamaan nilai solusi.



Gambar 3.1. Plotting hasil perhitungan bobot menggunakan metode LU (biru), QR (hijau), dan SVD (merah).

Berbeda dengan LU, QR, dan SVD, metode LMS menentukan bobot melalui proses pembelajaran iteratif. Pada kondisi awal, bobot LMS yang diinisialisasi nol (Grafik warna Ungu Gambar 3.1) menghasilkan nilai MSE sebesar 14.699,23164062. Nilai tersebut kemudian digunakan sebagai kondisi awal untuk proses pembelajaran. Dengan parameter laju pembelajaran  $\mu = 0,02$ , proses pemutakhiran bobot dilakukan secara iteratif menggunakan data pelatihan yang terdiri atas pasangan vektor input  $[m,n,o]$  dan nilai target  $d_k$ .



Gambar 3.2. Plotting MSE versus epoch pembelajaran LMS pada 5 epoch pertama (kiri), epoch-40 sampai epoch-140 (tengah), dan epoch-8047 (kanan).

Proses pembelajaran LMS, seperti diperlihatkan dalam Gambar 3.2, menghasilkan penurunan MSE yang sangat signifikan, bahkan sejak lima epoch awal. Setelah proses pembelajaran berlangsung hingga epoch ke-8047, diperoleh MSE sebesar 193,61979177. Dengan demikian, dibandingkan kondisi awal, proses pembelajaran menghasilkan penurunan MSE sebesar

$$\Delta MSE = 14.699,23164062 - 193,61979177 = 14.505,61184885.$$

Secara persentase, penurunan tersebut mencapai sekitar

$$\frac{14.505,61184885}{14.699,23164062} \times 100\% = 98,68\%.$$

Hasil ini menunjukkan bahwa proses pembelajaran LMS mampu memperbaiki solusi awal secara signifikan. Namun, nilai MSE akhir LMS belum digunakan untuk menyatakan metode tersebut sebagai metode terbaik atau terburuk, karena evaluasi tersebut perlu mempertimbangkan hasil metode LU, QR, dan SVD pada dataset yang sama.

Hasil penentuan vektor bobot dari keempat metode selanjutnya digunakan sebagai dasar untuk melakukan evaluasi komparatif. Untuk metode LU dan QR, evaluasi awal dilakukan dengan menghitung norma perbedaan masing-masing vektor terhadap vektor bobot SVD. Sementara itu, kualitas keluaran masing-masing solusi dievaluasi menggunakan MSE. Pendekatan ini memungkinkan penelitian membedakan antara perbedaan numerik pada vektor bobot dan perbedaan performa keluaran CMAC. Secara ringkas, hasil awal penentuan solusi dapat disajikan pada Tabel 3.1.

Tabel 3.1. Hasil perhitungan MSE solusi bobot CMAC

| Metode | Jumlah Bobot | Kondisi/Proses | MSE          |
|--------|--------------|----------------|--------------|
| SVD    | 1.529        | Solusi SVD     | 189,06238163 |
| LU     | 1.529        | Solusi LU      | 189,06238877 |
| QR     | 1.529        | Solusi QR      | 189,06238159 |
| LMS    | 1.529        | Epoch 8047     | 193,61979177 |

Tabel tersebut memperlihatkan bahwa nilai MSE dari LU, QR, dan SVD berada pada rentang yang sangat berdekatan, sedangkan MSE LMS pada epoch ke-8047 memiliki nilai yang sedikit lebih tinggi. Meskipun demikian, perbedaan MSE tersebut belum memberikan informasi mengenai seberapa berbeda vektor bobot yang dihasilkan oleh masing-masing metode. Oleh karena itu, tahap berikutnya dilakukan dengan membandingkan vektor bobot LU dan QR terhadap solusi SVD secara langsung melalui norma perbedaan vektor.

### 3.2 Evaluasi Performa Solusi LMS

Metode LMS memiliki karakteristik yang berbeda dibandingkan dengan metode LU, QR, dan SVD karena solusi bobot diperoleh melalui proses pembelajaran iteratif. Pada setiap proses pembelajaran, bobot diperbarui berdasarkan kesalahan antara keluaran CMAC dan nilai target. Oleh karena itu, evaluasi LMS dalam penelitian ini tidak hanya mempertimbangkan nilai MSE akhir, tetapi juga membandingkan kondisi sebelum dan sesudah proses pembelajaran untuk mengetahui tingkat perbaikan solusi yang dihasilkan.

Pada kondisi awal sebelum proses iterasi, solusi LMS menghasilkan nilai MSE sebesar 14.699,23164062. Nilai tersebut menunjukkan tingkat kesalahan yang masih tinggi sebelum bobot mengalami proses pemutakhiran berdasarkan data pelatihan. Dengan menggunakan parameter laju pembelajaran  $\mu = 0,02$ , proses pembelajaran dilakukan secara iteratif hingga diperoleh hasil pada epoch ke-8047. Pada epoch tersebut, MSE yang diperoleh mengalami penurunan menjadi 193,61979177. Perubahan MSE dari kondisi awal hingga epoch ke-8047 dapat dirangkum dalam Tabel 3.2 sebagai berikut.

Tabel 3.2. Perubahan MSE selama proses pembelajaran LMS

| Kondisi LMS          | Epoch | MSE             |
|----------------------|-------|-----------------|
| Sebelum iterasi      | 0     | 14.699,23164062 |
| Setelah pembelajaran | 8.047 | 193,61979177    |
| Penurunan MSE        | —     | 14.505,61184885 |
| Persentase penurunan | —     | 98,68%          |

Berdasarkan Tabel 3.2, proses pembelajaran LMS menghasilkan penurunan MSE sebesar 14.505,61184885, atau sekitar 98,68% dibandingkan kondisi awal. Penurunan tersebut

menunjukkan bahwa proses pemutakhiran bobot mampu memperbaiki kualitas solusi secara signifikan. Bahkan menurut Gambar 3.2, penurunan signifikan MSE tersebut terjadi di 5 epoch awal khususnya epoch pertama. Dengan demikian, LMS berhasil mengubah kondisi awal dengan tingkat kesalahan yang tinggi menjadi solusi dengan MSE yang jauh lebih rendah setelah proses pembelajaran berlangsung.

### 3.3 Perbandingan Keseluruhan Solusi

Berdasarkan hasil eksperimen pada subbagian sebelumnya, keempat metode yang digunakan menghasilkan solusi vektor bobot CMAC dengan karakteristik komputasi yang berbeda. Metode LU, QR, dan SVD memperoleh solusi melalui pendekatan penyelesaian sistem persamaan, sedangkan LMS memperoleh solusi melalui proses pembelajaran iteratif. Oleh karena itu, perbandingan keseluruhan dilakukan berdasarkan dua aspek utama, yaitu kedekatan numerik vektor bobot terhadap solusi SVD dan performa keluaran yang diukur menggunakan MSE.

Solusi SVD digunakan sebagai referensi dalam pengukuran kedekatan vektor bobot. Berdasarkan hasil komputasi, norma perbedaan vektor bobot LU terhadap SVD adalah

$$\|w_{LU} - w_{SVD}\|_2 = 2302,983421,$$

sedangkan untuk QR diperoleh

$$\|w_{QR} - w_{SVD}\|_2 = 980,6251922.$$

Pada solusi LMS yang diperoleh pada epoch ke-8047, norma perbedaan terhadap SVD adalah

$$\|w_{LMS} - w_{SVD}\|_2 = 38,91992.$$

Hasil tersebut menunjukkan perbedaan yang cukup jelas dalam kedekatan solusi terhadap SVD. Dari ketiga metode yang dibandingkan terhadap SVD, LMS pada epoch ke-8047 menghasilkan jarak vektor paling kecil, diikuti oleh QR dan kemudian LU. Dengan demikian, berdasarkan ukuran norma Euclidean, solusi LMS pada kondisi akhir eksperimen memiliki kedekatan numerik yang lebih tinggi terhadap solusi SVD dibandingkan solusi LU dan QR.

Perbandingan tersebut dapat dinyatakan melalui rasio jarak terhadap SVD. Jarak solusi LU sekitar

$$\frac{2302,983421}{38,91992} \approx 59,18$$

kali lebih besar daripada jarak solusi LMS. Sementara itu, jarak solusi QR sekitar

$$\frac{980,6251922}{38,91992} \approx 25,20$$

kali lebih besar daripada jarak solusi LMS. Hasil ini menunjukkan bahwa meskipun LU dan QR merupakan metode penyelesaian langsung, vektor bobot yang diperoleh keduanya dalam eksperimen ini memiliki perbedaan numerik yang jauh lebih besar terhadap solusi SVD dibandingkan solusi LMS pada epoch ke-8047.

Hasil pada Tabel 3.3 memperlihatkan temuan yang menarik ketika dibandingkan dengan nilai MSE. Berdasarkan kedekatan vektor bobot, LMS merupakan solusi yang paling dekat terhadap SVD pada epoch ke-8047. Namun, berdasarkan MSE, LMS masih menghasilkan nilai sebesar 193,61979177, sedangkan SVD menghasilkan 189,06238163. Dengan demikian, kedekatan vektor bobot terhadap SVD tidak secara langsung menghasilkan kesamaan nilai MSE.

Fenomena tersebut memperkuat hasil yang telah diperoleh sebelumnya, yaitu bahwa perbedaan numerik antar-vektor bobot tidak selalu berbanding lurus dengan perbedaan performa keluaran CMAC. LU memiliki jarak terhadap SVD yang jauh lebih besar daripada QR, tetapi MSE LU dan QR hampir identik dengan MSE SVD. Sebaliknya, LMS memiliki jarak terhadap SVD yang jauh lebih kecil, tetapi MSE-nya masih sedikit lebih tinggi.

Tabel 3.3. Perbandingan keseluruhan vektor bobot terhadap solusi SVD

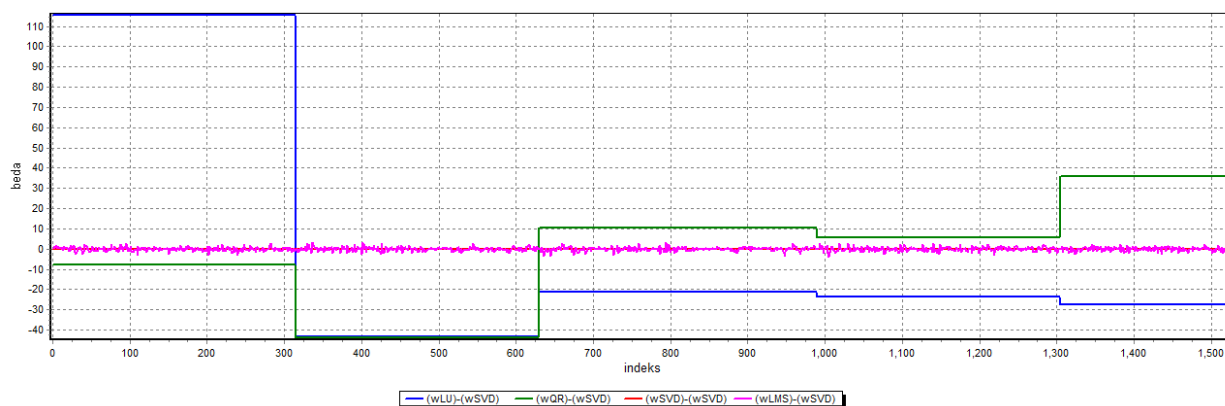
| Metode           | Norma Perbedaan terhadap SVD | MSE          | Keterangan                |
|------------------|------------------------------|--------------|---------------------------|
| SVD              | 0                            | 189,06238163 | Solusi referensi          |
| LU               | 2302,983421                  | 189,06238877 | Jarak terbesar dari SVD   |
| QR               | 980,6251922                  | 189,06238159 | Lebih dekat daripada LU   |
| LMS (epoch 8047) | 38,91992                     | 193,61979177 | Paling dekat terhadap SVD |

Dari sisi performa keluaran, SVD menghasilkan MSE sebesar 189,06238163, LU sebesar 189,06238877, QR sebesar 189,06238159, dan LMS pada epoch ke-8047 sebesar 193,61979177. Dengan demikian, MSE LMS lebih tinggi sebesar

$$193,61979177 - 189,06238163 = 4,55741014$$

dibandingkan SVD, atau sekitar 2,41% lebih tinggi. Meskipun demikian, nilai tersebut perlu dibandingkan dengan kondisi awal LMS yang memiliki MSE sebesar 14.699,23164062. Proses pembelajaran telah menurunkan MSE sebesar sekitar 98,68%, sehingga solusi LMS mengalami peningkatan performa yang sangat signifikan selama proses iteratif khususnya di lima epoch awal.

Selain hasil numerik tersebut, pengamatan selama proses iterasi LMS memperlihatkan bahwa nilai elemen-elemen vektor bobot LMS secara bertahap bergerak menuju nilai elemen bobot SVD yang bersesuaian seperti terlihat dalam Gambar 3.3. Temuan visual ini konsisten dengan hasil pengukuran norma pada epoch ke-8047 yang hanya sebesar 38,91992. Dengan demikian, terdapat indikasi yang kuat bahwa proses pembelajaran LMS dalam eksperimen ini mengarahkan vektor bobotnya semakin dekat terhadap solusi SVD.



Gambar 3.3. Visualisasi proses iterasi LMS yang menunjukkan kecenderungan elemen-elemen vektor bobot LMS (ungu) menuju nilai bobot SVD (merah) yang bersesuaian.

Meskipun demikian, pengamatan terhadap dinamika individual elemen bobot tidak menjadi fokus analisis kuantitatif pada artikel ini. Informasi tersebut digunakan sebagai bukti pendukung untuk menginterpretasikan kedekatan solusi LMS terhadap SVD. Analisis lebih rinci mengenai lintasan setiap elemen bobot, perbedaan laju konvergensi antar-elemen, serta hubungan antara pemutakhiran bobot dan proses konvergensi akan dibahas secara khusus dalam penelitian lanjutan.

Secara keseluruhan, hasil eksperimen memperlihatkan bahwa kedekatan vektor bobot dan performa MSE merupakan dua aspek evaluasi yang tidak selalu memberikan urutan yang sama. Berdasarkan norma terhadap SVD, urutan kedekatan solusi adalah

$$LMS < QR < LU$$

dengan SVD sebagai referensi. Sebaliknya, berdasarkan nilai MSE, urutan numerik dari nilai terkecil adalah

$$QR < SVD < LU < LMS$$

Namun, urutan MSE tersebut tidak tepat ditafsirkan sebagai peringkat performa yang substantif karena perbedaan antara QR, SVD, dan LU berada pada orde yang sangat kecil. Temuan yang lebih penting adalah bahwa LMS menghasilkan vektor bobot yang paling dekat terhadap SVD pada epoch ke-8047, sementara performa MSE-nya telah mendekati metode LU, QR, dan SVD setelah mengalami penurunan MSE sebesar 98,68% dari kondisi awal.

Dengan demikian, hasil investigasi menunjukkan bahwa LMS mampu menghasilkan solusi yang secara numerik semakin mendekati solusi SVD sekaligus memperbaiki performa keluaran secara signifikan. Di sisi lain, hasil LU dan QR memperlihatkan bahwa perbedaan vektor bobot yang relatif besar terhadap SVD tidak selalu menyebabkan perbedaan MSE yang berarti. Temuan tersebut menjadi dasar penting untuk memahami hubungan antara representasi vektor bobot dan performa keluaran CMAC.

### 3.4 Pembahasan Hasil Investigasi Komputasional

Hasil eksperimen menunjukkan bahwa metode LU, QR, SVD, dan LMS menghasilkan karakteristik solusi vektor bobot CMAC yang berbeda, meskipun seluruh metode diterapkan pada dataset dan struktur CMAC yang sama. Perbedaan tersebut terlihat jelas apabila solusi dibandingkan dari dua perspektif, yaitu kedekatan numerik vektor bobot terhadap solusi SVD dan performa keluaran yang diukur menggunakan MSE. Kedua perspektif tersebut memberikan informasi yang saling melengkapi mengenai karakteristik solusi yang dihasilkan.

Berdasarkan norma perbedaan terhadap solusi SVD, metode LU menghasilkan jarak sebesar 2302,983421, sedangkan QR menghasilkan jarak sebesar 980,6251922. Sementara itu, solusi LMS pada epoch ke-8047 menghasilkan jarak hanya sebesar 38,91992. Hasil tersebut menunjukkan bahwa pada kondisi akhir eksperimen, vektor bobot LMS merupakan solusi yang paling dekat secara numerik terhadap vektor bobot SVD dibandingkan dengan solusi LU dan QR. Kedekatan tersebut juga konsisten dengan pengamatan selama proses iterasi, ketika elemen-elemen bobot LMS secara visual terlihat bergerak menuju nilai elemen bobot SVD yang bersesuaian.

Temuan tersebut memberikan indikasi bahwa proses pembelajaran LMS tidak hanya memperbaiki kesalahan keluaran, tetapi juga mengarahkan solusi vektor bobot menuju karakteristik solusi yang diperoleh melalui SVD. Dengan kata lain, proses iteratif LMS dalam eksperimen ini memperlihatkan kecenderungan untuk menghasilkan representasi bobot yang semakin dekat dengan solusi SVD. Namun, hasil tersebut dipahami sebagai kecenderungan berdasarkan hasil eksperimen pada epoch yang diamati, bukan sebagai pembuktian matematis bahwa LMS akan identik dengan SVD pada kondisi iterasi tak hingga.

Fenomena yang menarik muncul ketika kedekatan vektor bobot dibandingkan dengan MSE. Meskipun LU memiliki norma perbedaan terhadap SVD sebesar 2302,983421 dan QR sebesar 980,6251922, nilai MSE keduanya hampir sama dengan SVD. SVD menghasilkan MSE 189,06238163, LU 189,06238877, dan QR 189,06238159. Selisih MSE LU terhadap SVD hanya 0,00000714, sedangkan selisih MSE QR terhadap SVD sekitar 0,00000004. Dengan demikian, perbedaan numerik yang relatif besar pada vektor bobot tidak menyebabkan perbedaan yang berarti pada MSE keluaran CMAC.

Hasil tersebut menunjukkan bahwa kedekatan dua solusi dalam ruang vektor bobot tidak selalu berbanding langsung dengan kedekatan performa keluarannya. Dalam konfigurasi CMAC yang digunakan, beberapa vektor bobot yang berbeda secara numerik dapat menghasilkan keluaran dengan tingkat kesalahan yang hampir sama. Kondisi ini penting dalam interpretasi hasil penelitian karena menunjukkan bahwa evaluasi solusi CMAC tidak cukup dilakukan hanya dengan membandingkan nilai setiap bobot atau jarak vektor terhadap suatu solusi referensi. Kualitas solusi

juga perlu dilihat dari kemampuan vektor bobot tersebut menghasilkan keluaran yang sesuai dengan target.

Karakteristik LMS memberikan perspektif tambahan. Pada kondisi awal, LMS menghasilkan MSE sebesar 14.699,23164062. Setelah proses pembelajaran hingga epoch ke-8047, MSE turun menjadi 193,61979177. Dengan demikian, proses pembelajaran menghasilkan penurunan MSE sebesar 14.505,61184885 atau sekitar 98,68%. Penurunan yang sangat besar tersebut menunjukkan bahwa proses pemutakhiran bobot LMS mampu memperbaiki solusi awal secara signifikan.

Meskipun MSE akhir LMS masih lebih tinggi dibandingkan SVD, selisihnya relatif kecil. MSE LMS sebesar 193,61979177 hanya 4,55741014 lebih tinggi daripada MSE SVD sebesar 189,06238163, atau sekitar 2,41% lebih tinggi. Hal tersebut menunjukkan bahwa solusi LMS telah mencapai tingkat performa yang dekat dengan solusi LU, QR, dan SVD. Pada saat yang sama, norma perbedaan LMS terhadap SVD yang hanya sebesar 38,91992 menunjukkan bahwa solusi LMS juga telah berada cukup dekat dengan solusi SVD dalam ruang vektor bobot.

Dengan demikian, hasil investigasi memperlihatkan adanya dua karakteristik evaluasi yang berbeda. Pertama, norma perbedaan memberikan informasi mengenai kedekatan representasi solusi dalam ruang vektor bobot. Kedua, MSE memberikan informasi mengenai performa solusi dalam menghasilkan keluaran CMAC. Kedua ukuran tersebut tidak selalu menghasilkan urutan yang sama. LMS merupakan contoh paling jelas: solusi LMS memiliki jarak paling kecil terhadap SVD, tetapi belum menghasilkan MSE paling rendah. Sebaliknya, LU memiliki jarak paling besar terhadap SVD di antara solusi yang dibandingkan, tetapi menghasilkan MSE yang hampir identik dengan SVD.

Perbedaan tersebut dapat dipahami dalam konteks representasi CMAC. Vektor bobot tidak bekerja sebagai kumpulan parameter yang berdiri sendiri, tetapi digunakan secara bersama-sama dalam pembentukan keluaran berdasarkan struktur dan pemetaan CMAC. Oleh karena itu, perubahan numerik pada sebagian bobot tidak harus menghasilkan perubahan keluaran yang sebanding. Selama kombinasi bobot yang digunakan oleh struktur CMAC mampu menghasilkan keluaran yang mendekati target, solusi dengan nilai bobot yang berbeda masih dapat memberikan MSE yang hampir sama.

Dari perspektif metode komputasi, hasil penelitian juga memperlihatkan perbedaan antara pendekatan langsung dan iteratif. LU, QR, dan SVD memperoleh solusi melalui proses komputasi langsung sehingga tidak memerlukan proses pembelajaran bertahap. Sebaliknya, LMS membutuhkan pemutakhiran bobot secara berulang untuk memperbaiki solusi. Dalam eksperimen ini, LMS berhasil melakukan perbaikan yang sangat besar dari kondisi awal dan pada epoch ke-8047 menghasilkan solusi yang secara numerik semakin dekat terhadap SVD serta memiliki performa MSE yang relatif dekat dengan metode lainnya.

Hasil ini menunjukkan bahwa LMS memiliki karakteristik yang berbeda dari metode dekomposisi matriks. Keunggulan LMS dalam konteks eksperimen bukan terletak pada pencapaian MSE terendah, tetapi pada kemampuannya memperoleh solusi melalui proses adaptif yang secara bertahap mendekati solusi referensi. Sebaliknya, LU dan QR mampu menghasilkan performa MSE yang sangat dekat dengan SVD meskipun solusi vektor bobotnya memiliki perbedaan numerik yang cukup besar terhadap SVD.

Secara keseluruhan, investigasi komputasional menghasilkan tiga temuan utama. Pertama, QR lebih dekat terhadap SVD dibandingkan LU berdasarkan norma perbedaan vektor, dengan nilai masing-masing 980,6251922 dan 2302,983421. Kedua, ketiga metode LU, QR, dan SVD menghasilkan MSE yang secara praktis hampir identik, sehingga perbedaan vektor bobot tidak secara langsung mencerminkan perbedaan performa keluaran. Ketiga, LMS pada epoch ke-8047 menghasilkan norma perbedaan terhadap SVD sebesar 38,91992 dan MSE sebesar 193,61979177, setelah sebelumnya menurunkan MSE awal sebesar 98,68%. Hasil tersebut menunjukkan bahwa proses iteratif LMS mampu membawa solusi bobot dan performa keluarannya mendekati karakteristik solusi SVD.

Dengan demikian, hasil investigasi tidak menunjukkan bahwa satu metode secara mutlak lebih unggul daripada metode lainnya. Setiap metode memiliki karakteristik solusi yang berbeda. SVD berperan sebagai solusi referensi dalam penelitian ini, LU dan QR menghasilkan solusi langsung dengan performa MSE yang hampir sama dengan SVD, sedangkan LMS menunjukkan kemampuan adaptif untuk memperbaiki solusi secara iteratif dan mendekatkan vektor bobotnya terhadap solusi SVD. Temuan tersebut memperlihatkan bahwa evaluasi metode penentuan bobot CMAC perlu mempertimbangkan baik kedekatan solusi vektor maupun performa keluaran, karena kedua aspek tersebut dapat memberikan informasi yang berbeda.

Berdasarkan hasil tersebut, penelitian ini memperlihatkan bahwa aplikasi demonstrasi yang dikembangkan dapat digunakan sebagai sarana investigasi komputasional terhadap perbedaan karakteristik solusi bobot CMAC. Aplikasi tidak hanya menghasilkan nilai akhir bobot, tetapi memungkinkan perbandingan solusi yang diperoleh melalui pendekatan dekomposisi matriks dan pembelajaran iteratif pada permasalahan yang sama. Hasil ini sekaligus memberikan dasar untuk penelitian lanjutan mengenai dinamika konvergensi elemen-elemen vektor bobot LMS terhadap solusi SVD, yang berada di luar ruang lingkup artikel ini.

#### 4. KESIMPULAN

Berdasarkan hasil investigasi komputasional terhadap penentuan bobot Cerebellar Model Articulation Controller (CMAC) menggunakan metode LU, QR, SVD, dan LMS, dapat disimpulkan bahwa keempat metode mampu menghasilkan solusi vektor bobot pada struktur CMAC yang digunakan, tetapi memiliki karakteristik komputasi dan kedekatan solusi yang berbeda. Metode LU, QR, dan SVD menghasilkan solusi melalui pendekatan penyelesaian langsung, sedangkan LMS memperoleh solusi melalui proses pembelajaran iteratif.

Hasil perbandingan vektor bobot terhadap solusi SVD menunjukkan bahwa norma perbedaan solusi LU adalah 2302,983421, sedangkan QR sebesar 980,6251922. Sementara itu, solusi LMS pada epoch ke-8047 memiliki norma perbedaan hanya sebesar 38,91992. Dengan demikian, pada kondisi akhir eksperimen, solusi LMS merupakan vektor bobot yang paling dekat secara numerik terhadap solusi SVD dibandingkan LU dan QR. Hasil pengamatan selama proses iterasi juga menunjukkan kecenderungan elemen-elemen bobot LMS bergerak menuju nilai elemen bobot SVD yang bersesuaian.

Ditinjau dari performa keluaran berdasarkan MSE, metode SVD menghasilkan MSE sebesar 189,06238163, LU sebesar 189,06238877, dan QR sebesar 189,06238159. Ketiga nilai tersebut menunjukkan bahwa solusi LU, QR, dan SVD menghasilkan performa yang secara praktis hampir identik, meskipun terdapat perbedaan numerik yang cukup besar pada vektor bobotnya. Temuan ini menunjukkan bahwa kedekatan vektor bobot terhadap solusi referensi tidak selalu berbanding langsung dengan perbedaan performa keluaran CMAC.

Pada metode LMS, MSE awal sebesar 14.699,23164062 berhasil diturunkan menjadi 193,61979177 pada epoch ke-8047, atau mengalami penurunan sekitar 98,68%. Meskipun MSE akhir LMS masih sekitar 2,41% lebih tinggi dibandingkan MSE SVD, hasil tersebut menunjukkan bahwa proses pembelajaran LMS mampu memperbaiki solusi secara signifikan dan menghasilkan performa yang mendekati metode LU, QR, dan SVD. Bersamaan dengan itu, norma LMS terhadap SVD yang relatif kecil menunjukkan bahwa proses pembelajaran juga mengarahkan vektor bobot LMS mendekati solusi SVD.

Secara keseluruhan, penelitian ini menunjukkan bahwa evaluasi solusi bobot CMAC sebaiknya tidak hanya didasarkan pada satu ukuran. Norma perbedaan vektor memberikan informasi mengenai kedekatan representasi bobot, sedangkan MSE memberikan informasi mengenai performa keluaran CMAC. Kedua ukuran tersebut dapat menghasilkan perspektif yang berbeda terhadap solusi yang diperoleh. Hasil investigasi juga memperlihatkan bahwa LMS mampu menghasilkan solusi yang mendekati solusi SVD melalui proses pembelajaran iteratif,

sedangkan LU dan QR mampu menghasilkan performa keluaran yang hampir sama dengan SVD meskipun vektor bobotnya memiliki perbedaan numerik yang lebih besar.

Penelitian selanjutnya dapat diarahkan pada analisis yang lebih mendalam terhadap dinamika konvergensi setiap elemen vektor bobot LMS, termasuk laju konvergensi, pola perubahan masing-masing bobot, serta hubungan antara proses pemutakhiran bobot dengan konvergensi terhadap solusi SVD. Analisis tersebut diharapkan dapat memberikan pemahaman lebih lanjut mengenai mekanisme konvergensi LMS dalam penentuan bobot CMAC.

## DAFTAR PUSTAKA

- Achmad, M. I., Susanto, A. dan Litasari 2004. Pemampatan Data Citra menggunakan CMAC. Tesis S2, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Gadjah Mada. <https://etd.repository.ugm.ac.id/penelitian/detail/23374>
- Achmad, M. I., Susanto, A. dan Adinugroho, H. 2014a. Pemetaan Alamat dan Fungsi Basis untuk Meningkatkan Unjuk-Kerja CMAC. Proceedings of 6TH National Conference on Information Technology and Electrical Engineering (CITEE) 2014, October 7 – 8 2014, Yogyakarta, Indonesia. <https://doi.org/10.5281/zenodo.7330500>
- Achmad, M. I., Adinugroho, H. dan Susanto, A. 2014b. Cerebellar Model Articulation Controller (CMAC) for Sequential Images Coding. Proceedings of 2014 1st International Conference on Information Technology, Computer and Electrical Engineering (ICITACEE), November 8 – 9 2014, Department of Electrical Engineering, Diponegoro University, Semarang, Indonesia.
- Achmad, M. I., Susanto, A. dan Adinugroho, H. 2015. Analisis Sintesis Citra Sekuensial berdasar Watak Citra melalui Rekayasa Cerebellar Model Articulation Controller (CMAC) menjadi Penyandi Laju Bit Rendah. Disertasi, S3 Ilmu Teknik Elektro, Universitas Gadjah Mada. <https://etd.repository.ugm.ac.id/penelitian/detail/79960>
- Achmad, M. I., Susanto, A. dan Adinugroho, H. 2014c. Weights Behavior in the Solution Space of the Cerebellar Model Articulation Controller (CMAC) learning. Proceedings of Regional Conference on Computer and Information Engineering (RCCIE), October 7 – 8 2014, Yogyakarta, Indonesia. <https://doi.org/10.5281/zenodo.7330539>
- Achmad, M. I., Sunantyo, T. A. dan Susanto, A. 2009. Cerebellar Model Associative Computer (CMAC) for Gravimetric Geoid Study based on EGM96 and EIGEN-GL04C Geopotential Development. Proceedings of Conference on Information Technology and Electrical Engineering (CITEE) 2009, International Session, August 4 2009, Department of Electrical Engineering, Faculty of Engineering, Gadjah Mada University, Yogyakarta, Indonesia. <https://doi.org/10.5281/zenodo.7330248>
- Achmad, M. I., (2021). Metode Doolittle untuk Faktorisasi LU. <https://doi.org/10.5281/zenodo.7330748>
- Achmad, M. I., (2005). Matriks GAUSS. Penerbit MAHANDIA. <https://doi.org/10.5281/zenodo.12699554>
- Albus, J. S. 1975a. A New Approach to Manipulator Control: The Cerebellar Model Articulation Controller, J. Dynamic Syst. Measurement Contr., September 1975, 220-227.
- Albus, J. S. 1975b. Data Storage in The Cerebellar Model Articulation Controller (CMAC). J. Dynam. Syst., Meas., Contr., September 1975, 228-233.
- Albus, J. S. 1971. A Theory of Cerebellar Function. Mathematical Bioscience, vol. 10, February 1971, 25-61.
- Borland, 2002. Borland Delphi Enterprise Version 7.0. Borland Software Corporation.
- Ker, J. S., Kuo, Y. H., Wen, R. C. dan Liu, B.D. 1997. Hardware Implementation of CMAC Neural Network with Reduced Storage Requirement. IEEE Trans. On Neural Network, Vol. 8, No. 6, November 1997, 1545-1556.
- Golub, G. H. dan Van Loan, C. F. 1989. Matrix Computations, Second Edition. The Johns Hopkins University Press, London.