

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

A. Nilai *Eigen*, Vektor *Eigen*, dan Dekomposisi Nilai Singular

Dalam mempelajari keadaan dinamik dari suatu sistem, khususnya sistem linear, dapat menggunakan sifat dari nilai eigen sistem dinamik tersebut. Berikut akan dijelaskan mengenai nilai *eigen*, vektor *eigen*, dan dekomposisi nilai singular.

1. Nilai *Eigen* (*Eigen Value*)

Eigen value atau nilai eigen (λ) merupakan nilai karakteristik suatu matriks yang menggambarkan matriks tersebut dalam bentuk perkalian dengan suatu vektor. Beberapa himpunan persamaan simultan dapat dinyatakan dalam bentuk matriks :

$$[A] \{X\} = \lambda \{X\} \dots\dots\dots (2.1)$$

dengan A merupakan matriks bujursangkar, X adalah vektor, dan λ adalah bilangan skalar. Dari persamaan (2.1), terlihat bahwa solusi dari persamaan tersebut adalah suatu vektor $\{X\}$ yang jika dikalikan oleh matriks [A], hasil kalinya adalah sama dengan perkalian suatu nilai skalar λ dengan $\{X\}$. Persamaan tersebut dapat ditulis dalam bentuk persamaan homogen sebagai :

$$([A] - \lambda [I]) \{X\} = 0 \dots\dots\dots (2.2)$$

dengan [I] adalah matriks identitas yang berordo sama dengan matriks [A]. Karena persamaan (2.2) homogen, maka persamaan tersebut konsisten dan selalu

mempunyai solusi trivial ($X=0$). Solusi non-trivial dari persamaan tersebut hanya ada jika nilai determinan matriks koefisien sama dengan nol.

$$|A - \lambda I| = \begin{vmatrix} A_{11} - \lambda & A_{12} & \dots & A_{1n} \\ A_{21} & A_{22} - \lambda & \dots & A_{2n} \\ \dots & \dots & \dots & \dots \\ A_{n1} & A_{n2} & \dots & A_{nn} - \lambda \end{vmatrix} = 0 \dots\dots\dots (2.3)$$

Nilai skalar λ agar didapatkan solusi non-trivial disebut nilai *eigen* (*eigen value*). Solusi untuk $\{X\}$ yang berkaitan dengan nilai *eigen* pada persamaan (2.2) disebut vektor *eigen* yang dapat diperoleh dengan menyelesaikan himpunan persamaan homogen yang sesuai. (Gere dan Weaver, 1987).

2. Vektor *Eigen* (*Eigen Vector*)

Setelah didapatkan nilai *eigen* dari suatu matriks, selanjutnya dapat dihitung vektor *eigen* yang berkaitan dengan nilai *eigen* tersebut. Vektor *eigen* dapat diperoleh dengan menyelesaikan himpunan persamaan homogen yang sesuai (lihat persamaan 2.1 dan 2.2). Berhubungan dengan setiap nilai *eigen* λ_i yang berbeda-beda, akan terdapat vektor *eigen* X_i yang tidak sama dengan nol. Vektor *eigen* adalah penyelesaian dari persamaan homogen yang didapatkan dengan mensubstitusikan nilai λ_i ke dalam persamaan (2.1) sebagai berikut (Gere dan Weaver, 1987) :

$$(A - \lambda_i) X_i = 0 \dots\dots\dots (2.4)$$

3. Dekomposisi Nilai Singular

Dekomposisi nilai singular/ *singular value decomposition (SVD)* merupakan suatu metode yang menguraikan suatu matriks menjadi dalam bentuk dua matriks. Sebagai contoh, untuk suatu matriks $[A]$ berukuran $m \times n$ dengan rank $r > 0$, maka penguraian dari matriks $[A]$ dinyatakan dalam bentuk (Baker, 2005) :

$$[A] = [U] [S] [V]^T \dots\dots\dots (2.5)$$

Rank r menyatakan banyaknya jumlah kolom atau baris yang saling independen dengan kolom atau baris lainnya dalam suatu matriks. $[U]$ (vektor singular kiri) merupakan matriks orthogonal berukuran $m \times r$, sedangkan $[V]$ (vektor singular kanan) merupakan matriks orthogonal dengan ukuran $n \times r$. $[S]$ adalah nilai singular, berupa matriks diagonal berukuran $r \times r$ yang elemen-elemen diagonalnya merupakan akar positif dari nilai eigen matriks $[A]$.

B. Getaran Bebas Tanpa Redaman

Suatu sistem yang mengalami getar bebas tanpa adanya redaman merupakan hal penting dalam analisis dinamik. Ditinjau persamaan gerak (Arfiadi, 2003b) :

$$\mathbf{M} \ddot{\mathbf{U}} + \mathbf{C} \dot{\mathbf{U}} + \mathbf{K} \mathbf{U} = \mathbf{F} \dots\dots\dots (2.6)$$

dengan \mathbf{M} = matriks massa, \mathbf{C} = matriks redaman, \mathbf{K} = matriks kekakuan, \mathbf{F} = vektor beban. Maka matriks redaman \mathbf{C} dan vektor beban \mathbf{F} pada pers. (2.6) sama dengan nol. Hal ini erat kaitannya dengan sifat unik / khusus dari setiap struktur. Bila struktur bergetar bebas tanpa redaman, maka suatu waktu terjadi pola / ragam

getaran tertentu dari bentuk perpindahan (*displacement*) lantai-lantai, sehingga perpindahan dapat ditulis sebagai:

$$\mathbf{U} = \varphi_i \sin (\omega_i t + \alpha_i) \dots\dots\dots (2.7)$$

lalu dengan menurunkan persamaan (2.7) dua kali, didapatkan:

$$\dot{\mathbf{U}} = \omega_i \varphi_i \cos (\omega_i t + \alpha_i) \dots\dots\dots (2.8)$$

$$\ddot{\mathbf{U}} = -\omega_i^2 \varphi_i \sin (\omega_i t + \alpha_i) = -\omega_i^2 \mathbf{U} \dots\dots\dots (2.9)$$

dengan substitusi persamaan (2.9) dan (2.7) pada persamaan (2.6) akan menghasilkan:

$$-\mathbf{M} \omega_i^2 \varphi_i \sin (\omega_i t + \alpha_i) + \mathbf{K} \varphi_i \sin (\omega_i t + \alpha_i) = 0$$

$$(\mathbf{K} - \omega_i^2 \mathbf{M}) \varphi_i \sin (\omega_i t + \alpha_i) = 0$$

atau

$$(\mathbf{K} - \omega_i^2 \mathbf{M}) \varphi_i = 0 \dots\dots\dots (2.10)$$

Persamaan (2.10) merupakan persamaan simultan homogen karena pada ruas kanan nilainya sama dengan nol. Persamaan tersebut dikenal sebagai permasalahan nilai *eigen* (*eigenvalue*). Jika persamaan ini diselesaikan, nilai φ_i tidak bisa didapatkan. Maka dari itu biasanya dilakukan normalisasi, misalnya dengan mengambil nilai relatif dari elemen φ_i . Dalam hal ini φ_i merupakan *mode shape vector* (vektor bentuk ragam) atau *eigenvector* struktur, bila bergetar bebas dengan frekuensi sudut alami sebesar ω_i . Nilai waktu getar alami (periode) untuk ragam getar yang ke-i dapat diperoleh dengan persamaan:

$$T = \frac{2\pi}{T_i} \dots\dots\dots(2.11)$$

Untuk sistem dengan N-derajat kebebasan, memiliki sejumlah N ragam getaran φ_i sampai dengan φ_N yang masing-masing mengalami getar dengan frekuensi lingkaran alami ω_i sampai dengan ω_N (atau T , sampai dengan T_N). Jadi,

$$\omega_1 \text{ berpasangan dengan } \varphi_1 = \begin{pmatrix} \varphi_{11} \\ \varphi_{21} \\ \dots \\ \varphi_{N1} \end{pmatrix}_1 = \begin{pmatrix} \varphi_{11} \\ \varphi_{21} \\ \dots \\ \varphi_{N1} \end{pmatrix}, \omega_2 \text{ dan pasangannya } \varphi_2 = \begin{pmatrix} \varphi_{12} \\ \varphi_{22} \\ \dots \\ \varphi_{N2} \end{pmatrix}_2 = \begin{pmatrix} \varphi_{11} \\ \varphi_{21} \\ \dots \\ \varphi_{N1} \end{pmatrix}, \text{ dan seterusnya.}$$

Vektor bentuk ragam $\varphi = \{ \varphi \}$ jika dikelompokkan menjadi matriks bentuk ragam Φ , yaitu :

$$\Phi = [\varphi_1 \varphi_2 \dots \varphi_N] = \begin{bmatrix} \varphi_{11} & \varphi_{12} & \dots & \varphi_{1N} \\ \varphi_{21} & \varphi_{22} & \dots & \varphi_{2N} \\ \dots & \dots & \dots & \dots \\ \varphi_{N1} & \varphi_{N2} & \dots & \varphi_{NN} \end{bmatrix} \dots\dots\dots(2.12)$$

Persamaan (2.10) dapat diselesaikan dengan mengambil determinan dari $(\mathbf{K} - \omega_i^2 \mathbf{M})$ sama dengan nol, atau dapat dituliskan :

$$|\mathbf{K} - \omega_i^2 \mathbf{M}| = 0 \dots\dots\dots(2.13)$$

Dari persamaan (2.13) akan didapatkan nilai-nilai ω_i yang bila disubstitusikan ke persamaan (2.10) akan diperoleh nilai-nilai ragam getaran $\{\varphi\}_i$ setelah sebelumnya dilakukan normalisasi.

C. Analisis Struktur dengan Metode Matriks

Analisis struktur dengan metode matriks dikenal dengan dua cara, yaitu metode kekakuan dan metode fleksibilitas. Berikut ini akan dijelaskan secara umum mengenai kedua metode tersebut.

1. Metode Kekakuan (*Stiffness Method*)

Seiring dengan berkembangnya teknologi di bidang IT, analisis struktur berdasarkan metode matriks kekakuan menjadi cocok dan populer dilakukan perhitungan dengan bantuan komputer. Hal ini karena prosedur analisis metode ini sangat terpolah dan sistematis sehingga mudah dilakukan pemrograman dengan komputer. Dengan demikian dapat mempermudah pengerjaan dan menghemat waktu dalam melakukan analisis struktur yang rumit. Langkah – langkah perhitungan pada metode matriks kekakuan secara umum sebagai berikut.

- 1) Membentuk matriks kekakuan elemen dan gaya ujung jepit.
- 2) Membentuk matriks kekakuan dalam koordinat global.
- 3) Menyelesaikan persamaan linier simultan untuk mencari nilai perpindahan (*displacement*) pada struktur
- 4) Menghitung gaya-gaya batang pada struktur

Dalam metode matriks kekakuan permasalahan-permasalahan yang terjadi didasarkan pada hubungan antarvariabel sebagai berikut:

1) *Compatibility* / Kesepadanan, yaitu hubungan antara deformasi batang dan perpindahan titik-kumpul.

2) *Contitutive equation*, yaitu hubungan antara deformasi batang dan gaya-gaya dalam yang bekerja pada struktur.

3) Hubungan antara beban luar dan gaya-gaya dalam. (Arfiadi, 2003).

2. Metode Metode Fleksibilitas (*Flexibility Method*)

Selain metode kekakuan, analisis struktur berdasarkan matriks lainnya adalah dengan metode matriks fleksibilitas. Berkebalikan dengan metode kekakuan yang mengambil lendutan pada titik bebas struktur sebagai besaran yang harus dicari, pada metode fleksibilitas diambil gaya sebagai besaran yang harus dicari. Sehubungan dengan hal tersebut, analisis dengan metode ini dimulai dengan kesetimbangan gaya-gaya yang bekerja pada struktur. Langkah-langkah yang menjadi dasar dari analisis ini adalah :

1) Kesetimbangan antara gaya luar dan gaya dalam suatu sistem struktur

2) Hubungan antara gaya dalam (*internal force*) dan deformasi elemen struktur

3) Kontinuitas dari deformasi atau kompatibiliti.(Supartono dan Boen, 1980).

3. Pendekatan berdasarkan Matriks Fleksibilitas

Keuntungan dalam deteksi kerusakan struktur berdasarkan pendekatan metode matriks fleksibilitas bila dibandingkan dengan metode matriks kekakuan yang didasarkan pada sifat-sifat ragam adalah sebagai berikut ini. Ditinjau persamaan gerak

$$\mathbf{M}\ddot{\mathbf{U}} + \mathbf{C}\dot{\mathbf{U}} + \mathbf{K}\mathbf{U} = \mathbf{F} \quad \dots\dots\dots(2.14)$$

Dengan menggunakan analisis ragam dan sifat orthogonalitas bentuk ragam, selanjutnya didapatkan

$$\mathbf{M}^* = \boldsymbol{\phi}^T \mathbf{M} \boldsymbol{\phi} \quad \dots\dots\dots(2.15)$$

dan

$$\mathbf{K}^* = \boldsymbol{\phi}^T \mathbf{K} \boldsymbol{\phi} \quad \dots\dots\dots(2.16)$$

dengan $\boldsymbol{\phi}$ = bentuk ragam dengan sembarang normalisasi. Kemudian dari sifat-sifat getaran bebas dapat diperoleh

$$\mathbf{K}^* - \omega^2 \mathbf{M}^* = \mathbf{0} \quad \dots\dots\dots(2.17)$$

atau

$$\omega^2 = \lambda = \mathbf{M}^{*-1} \mathbf{K}^* \quad \dots\dots\dots(2.18)$$

Dengan menggabungkan persamaan (2.3) dan (2.5) maka didapatkan

$$\boldsymbol{\phi}^T \mathbf{K} \boldsymbol{\phi} - \boldsymbol{\phi}^T \mathbf{M} \boldsymbol{\phi} \lambda = \mathbf{0} \quad \dots\dots\dots(2.19)$$

definisikan

$$\mathbf{v} = (\boldsymbol{\varphi}^T \mathbf{M} \boldsymbol{\varphi})^{1/2} = \mathbf{M}^{1/2} \dots\dots\dots (2.20)$$

dengan \mathbf{v} adalah matriks diagonal dengan indeks massa ternormalisasi pada diagonal utamanya.

Substitusikan persamaan (2.20) pada persamaan (2.19) didapatkan

$$\boldsymbol{\varphi}^T \mathbf{K} \boldsymbol{\varphi} - \mathbf{v}^2 \boldsymbol{\lambda} = \mathbf{0}$$

atau

$$\boldsymbol{\varphi}^T \mathbf{K} \boldsymbol{\varphi} = \mathbf{v}^T \boldsymbol{\lambda} \mathbf{v} \dots\dots\dots (2.21)$$

Dari persamaan (2.21) matriks kekakuan dapat diperoleh dari

$$\mathbf{K} = (\boldsymbol{\varphi}^T)^{-1} \mathbf{v}^T \boldsymbol{\lambda} \mathbf{v} \boldsymbol{\varphi}^{-1} \dots\dots\dots (2.22)$$

dari persamaan (2.1.7) diperoleh

$$(\mathbf{v}^{-2} \boldsymbol{\varphi}^T \mathbf{M}) \boldsymbol{\varphi} = \mathbf{I} \text{ dan } \boldsymbol{\varphi}^T (\mathbf{M} \boldsymbol{\varphi} \mathbf{v}^{-2}) = \mathbf{I} \dots\dots\dots (2.23)$$

sehingga

$$\boldsymbol{\varphi}^{-1} = (\mathbf{v}^{-2} \boldsymbol{\varphi}^T \mathbf{M}) \text{ dan } (\boldsymbol{\varphi}^T)^{-1} = (\mathbf{M} \boldsymbol{\varphi} \mathbf{v}^{-2}) \dots\dots\dots (2.24)$$

dengan menggunakan persamaan (2.24), persamaan (2.22) lalu menjadi

$$\mathbf{K} = \mathbf{M} \boldsymbol{\varphi} \mathbf{v}^{-1} \boldsymbol{\lambda} \mathbf{v}^{-1} \boldsymbol{\varphi}^T \mathbf{M} \dots\dots\dots (2.25)$$

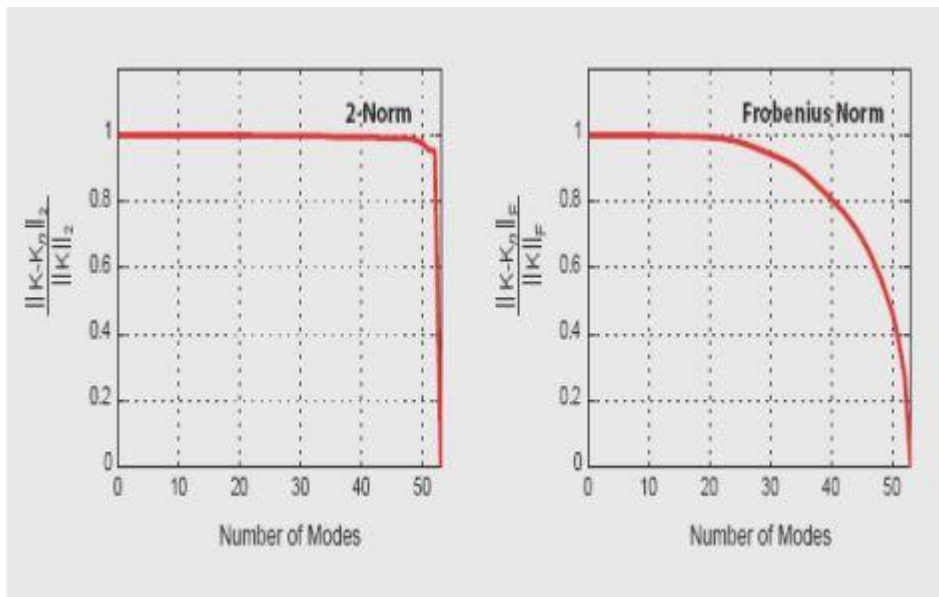
Karena hubungan antara matriks kekakuan dan matriks fleksibilitas, maka matriks fleksibilitas bisa didapatkan dari persamaan (2.22) sebagai

$$\mathbf{F} = (\boldsymbol{\varphi} \mathbf{v}^{-1}) \boldsymbol{\lambda}^{-1} (\boldsymbol{\varphi} \mathbf{v}^{-1})^T \dots\dots\dots (2.26)$$

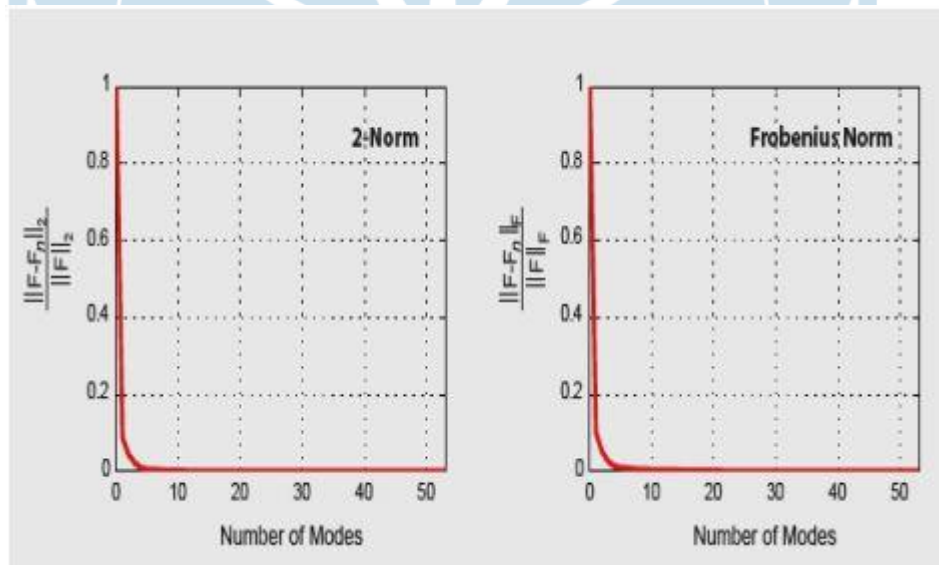
terlihat dari persamaan (2.25) dan (2.26) pengaruh setiap ragam frekuensi terhadap matriks kekakuan dan matriks fleksibilitas. Pengaruh ragam ke- j pada matriks kekakuan bertambah sebanding dengan pertambahan kuadrat frekuensi ragam ke- j . Sedangkan pengaruh ragam ke- j berkurang sebanding dengan kuadrat frekuensi ke- j .

Hal ini menunjukkan bahwa matriks kekakuan sensitif terhadap frekuensi ragam yang lebih tinggi, sedangkan matriks fleksibilitas tidak begitu sensitif. Dengan demikian identifikasi matriks fleksibilitas akan lebih menguntungkan dibandingkan dengan matriks kekakuan karena ragam-ragam yang lebih tinggi biasanya sukar dideteksi dalam percobaan di lapangan (Arfiadi dan Wibowo, 2005).

Gao (2005) melakukan analisis numerik pada suatu struktur rangka batang untuk menunjukkan bahwa pendekatan dengan matriks fleksibilitas tidak begitu sensitif terhadap ragam-ragam yang lebih tinggi, sedangkan pendekatan dengan matriks kekakuan memerlukan data dari ragam struktur yang lebih tinggi. Analisis ini menggunakan sistem rangka batang (*truss*) dengan 53 derajat kebebasan sebagai objek pengamatan dengan menggunakan 2-norm dan Frobenius norm. Berikut ini ditunjukkan dalam Gambar 2.1 dan 2.2.



Gambar 2.1 Kesalahan pada matriks kekakuan vs jumlah ragam



Gambar 2.2 Kesalahan pada matriks fleksibilitas vs jumlah ragam

D. Metode Deteksi Kerusakan Struktur

Metode yang digunakan untuk mendeteksi kerusakan pada struktur telah lama dikembangkan. Dalam tulisannya, Doebling dkk (1996) telah meninjau berbagai metode tersebut. Secara umum, metode-metode yang digunakan berdasarkan pada getaran (*vibration-based*). Metode-metode ini dikategorikan berdasarkan pada jenis data yang diukur dan/atau teknik yang digunakan untuk mengidentifikasi kerusakan.

Salah satu kategori yang dikembangkan adalah berdasarkan pada perubahan frekuensi (*frequency changes*). Adams et al. (1978) dan Cawley dan Adams (1979) mengusulkan metode deteksi lokasi kerusakan berdasarkan sensitivitas menggunakan perhitungan pada frekuensi. Metode ini berdasarkan asumsi rasio perubahan frekuensi antara 2 ragam adalah fungsi dari lokasi kerusakan. Dengan menghitung rasio frekuensi teoritis akibat kerusakan pada lokasi yang ditentukan dan membandingkannya dengan yang lain, lokasi kerusakan dapat ditentukan. Namun demikian didapatkan beberapa kesalahan hasil ketika menggunakan metode ini. Sebagai tambahan, tingkat ketepatan metode ini sangat bergantung pada akurasi model analitik. Stubbs (1985) dan Stubbs dan Osegueda (1987) mengusulkan metode yang berhubungan dengan perubahan kekakuan elemen terhadap perubahan frekuensi ragam getar. Metode ini berasumsi matriks kekakuan struktur dapat dikembangkan dalam bentuk matriks kekakuan elemen diagonal. Asumsi ini hanya dapat berlaku untuk struktur sederhana, seperti rangka batang. Kemudian dengan menghilangkan asumsi ini, Stubbs dan Osegueda (1990a, b) memberikan bentuk yang lebih umum untuk hubungan antara matriks kekakuan elemen dan frekuensi

ragam getar. Untuk beberapa skenario kerusakan, masih ditemukan kesalahan prediksi lokasi kerusakan menggunakan metode ini.

Kategori lainnya adalah menggunakan perubahan pada ragam getar (*mode shapes*). West (1984) menggunakan modal assurance criteria (MAC) untuk menentukan lokasi kerusakan struktur. Dalam metode ini, beberapa sensor diterapkan pada struktur. Setelah dilakukan studi numerik dan ekperimental terhadap balok, Fox (1992) menunjukkan jika nilai MAC tidak begitu sensitif/peka terhadap kerusakan dan menyarankan perbandingan secara grafik dari ragam getar untuk menentukan lokasi kerusakan. Dalam penelitian lain, Kim et al. (1992) menggunakan Coordinate MAC (COMAC) dalam hubungannya dengan Partial MAC (PMAC) untuk menentukan kerusakan pada struktur. Shi et al. (2000a) menggunakan *incomplete mode shape* untuk memperkirakan kerusakan struktur.

Energi regangan ragam getar (*modal strain energy*) juga dapat digunakan sebagai metode untuk mendeteksi kerusakan struktur. Dalam penelitiannya, Shi et al. (1998,2000) menunjukkan elemen dengan rasio perubahan energi regangan ragam getar yang tinggi dapat diidentifikasi sebagai elemen yang kemungkinan mengalami kerusakan. Kemudian dikembangkan algoritma kuantifikasi kerusakan dengan menyatakan perubahan energi regangan ragam getar dalam bentuk kontribusi dari elemen yang kemungkinan mengalami kerusakan. Dalam algoritma ini, diperlukan iterasi untuk mendapatkan perhitungan kerusakan yang konvergen, dan hasilnya menunjukkan bahwa diperlukan banyak ragam getar untuk mendapatkan perkiraan yang akurat dari tingkat kerusakan. Lalu Shi et al. (2002) mengembangkan algoritma ini dengan hanya menggunakan beberapa ragam getar.

Metode lain yang dikembangkan untuk mendeteksi kerusakan struktur adalah menggunakan gaya sisa (*residue force*) seperti yang dilakukan oleh Ricles dan Kosmatka (1992) dan Sheinmann (1996). Peneliti lainnya menggunakan metode estimasi parameter tidak hanya untuk menentukan lokasi kerusakan, namun juga untuk menghitung tingkat kerusakan yang terjadi (Yao et al, 1992).

Dalam penelitian yang lain, dilakukan pendekatan menggunakan perubahan matriks fleksibilitas. Metode ini berdasarkan pada relasi invers antara matriks fleksibilitas dan kekakuan. Adanya kerusakan dalam struktur, yang mana akan membuat turunnya tingkat kekakuan struktur, akan meningkatkan fleksibilitasnya. Pandey dan Biswas (1994, 1995b) menggunakan matriks fleksibilitas untuk menentukan lokasi kerusakan. Langkah pertama adalah dengan membentuk matriks fleksibilitas pada kondisi sebelum dan sesudah terjadi kerusakan. Nilai maksimum setiap kolom (setiap kolom yang berhubungan dengan suatu derajat kebebasan) dari perubahan matriks fleksibilitas (F_{Δ}) tersebut kemudian dipilih sebagai indikator lokasi kerusakan. Kolom dengan nilai yang lebih besar kemungkinan merupakan lokasi dimana kerusakan terjadi. Namun demikian, menggunakan perubahan matriks fleksibilitas secara langsung untuk menentukan lokasi kerusakan mungkin akan sulit dilakukan terhadap beberapa kasus. Kemudian Pandey dan Biswas (1995a) mengembangkan metode ini untuk menentukan lokasi dan besarnya kerusakan pada struktur. Dikembangkan formulasi untuk mengekspresikan hubungan antara perubahan matriks fleksibilitas sebagai fungsi dari perubahan matriks kekakuan elemen. Lalu diterapkan teknik kebalikan semu (*pseudo invers*) untuk mendapatkan perubahan kekakuan elemen. Toksoy dan Aktan (1994)

menghitung kerusakan pada jembatan berdasarkan perhitungan pada matriks fleksibilitas ragam getar. Dihitung perbedaan matriks fleksibilitas ragam getar pada kondisi sebelum dan sesudah terjadi kerusakan. Kemudian dilakukan *input* pembebanan pada matriks tersebut dengan konfigurasi beban yang berbeda-beda sehingga didapatkan defleksi pada profil yang digunakan. Lokasi kerusakan pada jembatan didapatkan dari perbandingan defleksi profil yang terjadi. Yang terbaru, Bernal (2002) mengusulkan metode vektor beban penentu lokasi kerusakan / *damage locating vector (DLV)*. Metode ini kemudian juga telah divalidasi dengan uji eksperimental (Gao et al. 2004).

E. Vektor Beban Penentu Lokasi Kerusakan (VBPLK)

1. Penelitian Terdahulu

Vektor beban penentu lokasi kerusakan merupakan salah satu metode untuk mendeteksi kerusakan struktur berdasarkan perubahan pada matriks fleksibilitas yang diusulkan oleh Bernal (2002). Prinsip dari teknik ini adalah menentukan suatu set vektor beban yang apabila dikerjakan sebagai beban statik pada lokasi sensor yang digunakan untuk pengukuran akan menghasilkan tegangan yang sangat kecil (mendekati nol) pada elemen yang rusak. Dengan demikian lokasi elemen yang dideteksi mengalami kerusakan dapat ditentukan.

Berdasarkan beberapa penelitian sebelumnya, metode terbukti efektif dan memiliki tingkat akurasi yang tinggi dalam menentukan lokasi kerusakan dari suatu sistem struktur. Aditya Kaushik, dkk (2014) melakukan studi numerik terhadap pendekatan VBPLK untuk deteksi kerusakan struktur menggunakan data beban

statik. Beberapa tipe struktur seperti *truss* 2D & 3D, balok *single span & multiple span*, dimodelkan dan dilakukan analisis untuk beberapa variasi skenario kerusakan. Hasil dari penelitian tersebut menunjukkan bahwa 94% kasus kerusakan dapat diidentifikasi menggunakan metode VBPLK ini.

Arfiadi dan Wibowo (2005) membahas mengenai deteksi kerusakan struktur dengan pendekatan matriks fleksibilitas. Metode vektor beban penentu lokasi kerusakan (VBPLK) diterapkan untuk mendeteksi adanya kerusakan pada struktur rangka batang. Matriks fleksibilitas struktur dibentuk pada kondisi rusak dan tidak rusak, kemudian dilakukan analisis *singular value decomposition* pada selisih kedua matriks fleksibilitas tersebut. Beban yang menghasilkan *singular value* sama dengan nol (atau nilainya sangat kecil) merupakan beban yang dapat mendeteksi kerusakan struktur. Berdasarkan pengamatan pada lokasi kerusakan, terlihat bahwa metode VBPLK dapat mendeteksi lokasi kerusakan pada struktur.

Richard Frans, dkk (2017) melakukan studi perbandingan metode deteksi kerusakan struktur antara metode *mode shapes curvature* dan metode vektor beban penentu lokasi kerusakan (VBPLK). Tiga tipe struktur digunakan sebagai studi kasus pada penelitian ini, yaitu *shear building*, balok, dan yang terakhir sistem portal bidang. Masing-masing tipe struktur dilakukan simulasi model numerik menggunakan beberapa skenario kerusakan untuk kedua metode tersebut. Berdasarkan penelitian tersebut, kedua metode terbukti memiliki sensitivitas yang baik dalam mendeteksi kerusakan struktur. Pada tipe *shear building* dan struktur balok kedua metode dapat memprediksi lokasi kerusakan. Namun pada portal bidang, metode VBPLK dapat memprediksi *member* yang mengalami kerusakan,

sedangkan metode *mode shape curvature* hanya dapat memprediksi nodal di sekitar *member* portal bidang yang mengalami kerusakan.

Gao, dkk (2007) melakukan verifikasi ekperimental terhadap metode vektor beban penentu lokasi kerusakan. Penelitian ini menggunakan struktur *truss* 3D dengan panjang 15 ft. Untuk mensimulasikan kerusakan pada struktur, salah satu batang pada *truss* tersebut diganti dengan batang yang kekakuannya sudah direduksi. Pada eksperimen ini digunakan 2 skenario kerusakan, yang pertama adalah pengurangan 40% kekakuan pada elemen longitudinal, dan yang ke dua adalah pengurangan 40% kekakuan pada elemen vertikal. Didapatkan kesimpulan bahwa dengan menggunakan metode VBPLK berdasarkan fleksibilitas, lokasi kerusakan pada struktur *truss* tersebut dapat ditentukan dengan benar hanya dengan menggunakan jumlah sensor yang terbatas.

Arfiadi (2012) menyelidiki mengenai pengaruh jumlah sensor pada deteksi kerusakan struktur dengan metode VBPLK. Penelitian ini menggunakan suatu sistem jembatan rangka sebagai studi kasus simulasi numerik untuk menunjukkan efektivitas penggunaan metode VBPLK dalam mendeteksi kerusakan sturktur. Variasi jumlah pengukuran dilakukan. Berdasarkan analisis diperoleh hasil bahwa semakin banyak sensor yang digunakan, jumlah batang yang terdeteksi sebagai kandidat batang yang rusak juga semakin banyak. Pada penelitian ini terlihat juga bahwa untuk struktur dengan jumlah derajat kebebasan sebesar 25, jumlah sensor 2 buah dapat mendeteksi lokasi kerusakan batang.

Pengembangan tipe struktur yang digunakan sebagai bahan penelitian terkait dengan metode VBPLK juga banyak dilakukan. Penelitian yang dilakukan Saputro

(2011) menunjukkan bahwa metode VBPLK dapat digunakan untuk mendeteksi kerusakan struktur baik kolom maupun balok pada portal bidang baja 6 lantai. Sedangkan penelitian Subiyanto (2013) menunjukkan hasil penggunaan metode VBPLK dalam mendeteksi kerusakan bracing pada portal bidang baja tipe bracing konsentrik dapat dilakukan.. Selain itu, penelitian ini juga mendapatkan kesimpulan variasi jumlah dan penempatan sensor memberikan pengaruh yang cukup signifikan terhadap hasil deteksi.

Berdasarkan penelitian-penelitian sebelumnya, penulis tertarik untuk menggunakan metode VBPLK untuk melakukan deteksi kerusakan struktur sampai dengan tingkat 3, yaitu tidak hanya menentukan lokasi elemen yang mengalami kerusakan namun juga memprediksi penurunan kekuatan/kekakuan struktur yang mengalami kerusakan.

2. Tinjauan Umum

Seperti yang telah disampaikan sebelumnya, metode vektor beban penentu lokasi kerusakan (VBPLK) didasarkan pada perbedaan matriks fleksibilitas pada lokasi sensor struktur pada kondisi sebelum dan sesudah mengalami kerusakan. Jika matriks fleksibilitas struktur sebelum dan sesudah mengalami kerusakan berturut-turut dapat disimbolkan dengan F_u dan F_d dan vektor beban pada lokasi sensor dinyatakan dengan L memenuhi persamaan berikut (Bernal,2002) :

$$F_u L = F_d L \dots\dots\dots (2.27)$$

atau

$$F_{\Delta} L = (F_d - F_u) L = 0 \dots\dots\dots(2.28)$$

Dari persamaan di atas, terlihat bahwa vektor beban L menghasilkan perpindahan (*displacement*) yang sama pada lokasi sensor untuk kondisi sebelum dan sesudah terjadi kerusakan struktur. Karena vektor beban penentu lokasi kerusakan (VBPLK) tidak menghasilkan tegangan pada elemen yang rusak, maka kerusakan elemen tersebut tidak mempengaruhi perpindahan pada tempat yang diukur (Arfiadi dan Wibowo, 2005). Maka vektor beban L adalah VBPLK.

3. Menentukan Vektor Beban Penentu Lokasi Kerusakan

Vektor beban penentu lokasi kerusakan dapat ditentukan dari hasil analisis *singular value decomposition* (SVD) terhadap nilai perbedaan matriks fleksibilitas (F_{Δ}) seperti berikut ini :

$$F_{\Delta} = U S V^T \dots\dots\dots(2.29)$$

atau

$$F_{\Delta} = [U_1 \ U_0] \begin{bmatrix} S_1 & 0 \\ 0 & 0 \end{bmatrix} [V_1 \ V_0]^T \dots\dots\dots(2.30)$$

Dengan mengingat sifat SVD

$$[V_1 \ V_0]^T [V_1 \ V_0] = \mathbf{I} \dots\dots\dots(2.31)$$

Persamaan tersebut dapat ditulis ulang menjadi

$$[F_{\Delta} V_1 \ F_{\Delta} V_0] = [U_1 \ S_1 \ 0] \dots\dots\dots(2.32)$$

sehingga

$$F_{\Delta} V_0 = 0 \dots\dots\dots(2.33)$$

Persamaan (2.28) dan (2.33) mengindikasikan jika $L = V_0$. Maka vektor beban penentu lokasi kerusakan dapat diperoleh dari nilai SVD perbedaan matriks fleksibilitasnya (F_{Δ}).

Pada persamaan (2.30), karena kesalahan dalam perhitungan dan pengukuran matriks fleksibilitas, nilai *singular value* yang berhubungan dengan V_0 secara umum tidak sama dengan nol. Untuk menentukan VBPLK dari SVD matriks F_{Δ} , Bernal (2002) mengusulkan digunakannya indeks *svn* sebagai berikut

$$svn = \sqrt{\frac{S_i C_i^2}{\max_k (S_k C_k^2)}} \dots\dots\dots(2.34)$$

dengan s_i = nilai *singular value* yang ke- i dari matriks F_{Δ} , c_i = konstanta yang digunakan untuk normalisasi tegangan maksimum elemen struktur, yang dipengaruhi oleh beban statik $c_i V_i$, sehingga bernilai sama dengan 1; dan V_i = *right singular vector* dari matriks F_{Δ} . Lebih lanjut, Bernal (2002) menyatakan untuk nilai indeks $svn \leq 0,20$ menunjukkan bahwa vektor beban merupakan VBPLK.

4. Normalisasi Tegangan Kumulatif

Setiap VBPLK yang diperoleh, dikerjakan pada model struktur yang tidak rusak. Tegangan pada setiap elemen struktur dapat diperoleh yang kemudian dapat dihitung tegangan kumulatif ternormalisasinya (*normalized cumulative stress*). Jika elemen mempunyai nilai tegangan kumulatif ternormalisasi sama dengan atau mendekati nol, maka elemen ini merupakan kandidat elemen yang rusak. (Arfiadi

dan Wibowo, 2005). Tegangan kumulatif ternormalisasi dapat ditentukan berdasarkan (Gao, 2005)

$$\bar{\sigma}_j = \frac{\sigma_j}{\max_k(\sigma_{ik})} \text{ dengan } \sigma_j = \sum_{i=1}^m \text{abs} \left(\frac{\sigma_{ij}}{\max_k(\sigma_{ik})} \right) \dots\dots\dots(2.35)$$

dengan σ_j = tegangan kumulatif pada elemen ke-j, σ_{ij} = tegangan pada elemen ke-j yang disebabkan oleh VBPLK ke-i, m = jumlah VBPLK.

5. Ringkasan Langkah untuk Mendeteksi Kerusakan Struktur menggunakan Metode VBPLK

Berdasarkan penjelasan di atas, langkah-langkah untuk melakukan deteksi kerusakan struktur menggunakan metode VBPLK seperti yang diusulkan Bernal (2002) secara sederhana dapat dijelaskan sebagai berikut :

- 1) Hitung perbedaan matriks fleksibilitas struktur ($F\Delta$)
- 2) Lakukan analisis nilai SVD terhadap matriks $F\Delta$
- 3) Ambil V_i dari V
- 4) Hitung tegangan pada struktur yang tidak rusak dengan V_i
- 5) Tentukan nilai konstanta c_i
- 6) Hitung indeks svn . Jika nilai $svn \leq 0,20$, maka vektor tersebut merupakan vektor beban penentu lokasi kerusakan (VBPLK).
- 7) Hitung nilai tegangan kumulatif ternormalisasinya
- 8) Untuk tegangan kumulatif ternormalisasi yang nilainya sama dengan atau mendekati nol, maka elemen tersebut merupakan kandidat elemen yang rusak.

F. *Modal Assurance Criterion (MAC)*

Dalam rekayasa struktur, analisis modal diperlukan untuk menentukan berbagai waktu getar di mana struktur tersebut akan beresonansi secara alami. Waktu getar adalah hal penting dalam teknik rekayasa gempa, karena frekuensi alami bangunan tidak sesuai dengan frekuensi gempa yang diharapkan di wilayah bangunan didirikan. Jika frekuensi alami struktur sesuai dengan frekuensi getar, struktur dapat terus beresonansi dan mengalami kerusakan. Analisis modal dalam struktur meliputi frekuensi alami, vektor ragam getar, modal massa, modal kinetik, dan energi regangan.

Modal Assurance Criterion (MAC) merupakan salah satu metode yang populer digunakan untuk mengukur perbedaan dari dua ragam getar. MAC pada awalnya digunakan untuk mengukur kualitas dari vektor modal yang didapatkan dari percobaan eksperimental dengan fungsi respon frekuensi yang diukur (Allemang, 2003). MAC adalah pengukuran statistik antara vektor modal (*modal vector*) yang diestimasi dengan vektor modal riil (*real modal vector*). Nilai MAC dapat dituliskan sebagai:

$$MAC_i = \frac{|\{\varphi_a\}_i^T \{\varphi_d\}_i|^2}{(\{\varphi_a\}_i^T \{\varphi_a\}_i)(\{\varphi_d\}_i^T \{\varphi_d\}_i)} \dots\dots\dots (2.36)$$

Karena dalam penelitian ini nilai MAC digunakan untuk membandingkan ragam getar struktur prediksi dan struktur yang mengalami kerusakan, maka dapat didefinisikan $\{\varphi_a\}$ sebagai ragam getar ke-i struktur prediksi dan $\{\varphi_d\}_i$ sebagai ragam getar aktual dari struktur yang mengalami kerusakan.

Nilai MAC berkisar antara 0 sampai dengan 1. Jika ragam getar struktur prediksi dan struktur aktual identik, maka nilai MAC akan mendekati atau sama dengan 1, sebaliknya jika ragam getar tidak identik (tidak berkorelasi sama sekali), nilai MAC akan mendekati atau sama dengan 0.

G. *Symbiotic Organism Search (SOS) Algorithm*

Min-Yuan Cheng dan Doddy Prayogo pada tahun 2014 mengusulkan algoritma SOS untuk menyelesaikan persoalan-persoalan desain teknik yang menyangkut optimasi numerik. Algoritma ini merupakan metode optimasi methaheuristik yang mensimulasikan interaksi simbiosis yang dilakukan oleh berbagai macam makhluk hidup/organisme di dalam suatu ekosistem, yang bertujuan untuk meningkatkan tingkat kelangsungan hidup dari organisme yang bersangkutan (Cheng dan Prayogo, 2014). Hubungan simbiosis antarorganisme secara umum dapat digolongkan menjadi 3 macam, yaitu :

1) Simbiosis mutualisme, yaitu hubungan antarorganisme yang saling menguntungkan satu sama lain. Contoh : lebah dengan bunga, burung bangau dan kudaniil

2) Simbiosis komensalisme, yaitu hubungan antarorganisme di mana salah satu organisme diuntungkan sedangkan organisme yang lain tidak diuntungkan namun juga tidak dirugikan. Contoh : ikan hiu dan ikan remora

3) Simbiosis parasitisme, yaitu hubungan antarorganisme di mana salah satu organisme diuntungkan, namun organisme lain dirugikan. Contoh : manusia dan virus, sapi dan cacing hati.

Setiap organisme, yang melambangkan variabel solusi, akan saling berinteraksi antar satu dengan yang lainnya secara acak dan akan mengalami ketiga simbiosis tersebut. Secara umum langkah kerja algoritma SOS terdiri dari 3 tahap/fase, yaitu : (1) Fase inialisasi; (2) Fase simbiosis (mutualisme, komensalisme, dan parasitisme); (3) Pengulangan sampai mendapatkan hasil yang diharapkan. Berikut ini *pseudo-code* untuk menggambarkan algoritma SOS secara garis besar (Prayogo, dkk, 2018) :

- 1: Inialisasi parameter
- 2: Inialisasi ekosistem dengan variabel acak
- 3: **while** (belum memenuhi kriteria berhenti) **do**
- 4: **for** masing-masing organisme i
- 5: %% *Mutualism phase*
- 6: pilih organisme j secara acak dari ekosistem
- 7: *update* solusi terbaik dalam ekosistem
- 8: modifikasi organisme i dan j melalui *mutualism phase*
- 9: evaluasi nilai objektif dari organisme i dan j setelah interaksi mutualisme

- 10: terima hasil modifikasi organisme i dan j apa bila nilai objektif yang dihasilkan lebih baik
- 11: %% *Commensalism phase*
- 12: pilih organisme j secara acak dari ekosistem
- 13: modifikasi organisme i melalui *commensalism phase*
- 14: evaluasi nilai objektif dari organisme i setelah interaksi komensalisme
- 15: terima hasil modifikasi organisme i apabila nilai objektif yang dihasilkan lebih baik
- 16: %% *Parasitism phase*
- 17: pilih organisme j secara acak dari ekosistem
- 18: bentuk "Parasite_Vector" melalui *parasitism phase*
- 19: evaluasi nilai objektif dari Parasite_Vector
- 20: ubah organisme j menjadi Parasite_Vector apabila nilai objektif yang dihasilkan lebih baik
- 21: **end for**
- 22: **end while**

Untuk rincian lebih lanjut mengenai ketiga fase ini akan dijabarkan pada sub-bab berikut.

1. *Mutualism Phase*

Dalam fase ini diambil dua organisme acak X_i dan X_j untuk saling berinteraksi. Kandidat solusi yang baru, yaitu new_X_i dan new_X_j , dihitung berdasarkan simbiosis mutualisme antara organisme X_i dan X_j . Simbiosis tersebut dimodelkan dalam persamaan (2.37) dan (2.38) berikut ini :

$$new_X_i = X_i + \text{rand}(0,1) * [X_{best} - \text{mean}(X_i, X_j)] * \text{round}[1 + \text{rand}(0,1)] \dots \dots (2.37)$$

$$new_X_j = X_j + \text{rand}(0,1) * [X_{best} - \text{mean}(X_i, X_j)] * \text{round}[1 + \text{rand}(0,1)] \dots \dots (2.38)$$

di mana X_{best} merupakan organisme terbaik dalam ekosistem. Organisme X_i dan X_j diperbarui hanya jika nilai objektif dari new_X_i dan new_X_j lebih baik daripada nilai objektif X_i dan X_j sebelum interaksi.

2. *Commensalism Phase*

Sama halnya dengan fase mutualisme, pada fase ini organisme X_i dipilih secara *random* dari ekosistem untuk berinteraksi dengan X_j . Kemudian dihitung kandidat solusi baru new_X_i sesuai dengan simbiosis komensalisme antara X_i dan X_j dengan menggunakan persamaan :

$$new_X_i = X_i + \text{rand}(-1,1) * (X_{best} - X_j) \dots \dots \dots (2.39)$$

Organisme X_i diperbaharui hanya jika nilai objektif dari new_X_i lebih baik daripada nilai objektif X_i sebelum interaksi.

3. *Parasitism Phase*

Dalam fase parasitisme, organisme X_i diberi peran untuk memproduksi parasit buatan yang disebut dengan “*Parasite_Vector*”. *Parasite_vector* adalah modifikasi acak dari X_i yang dimodelkan dalam persamaan :

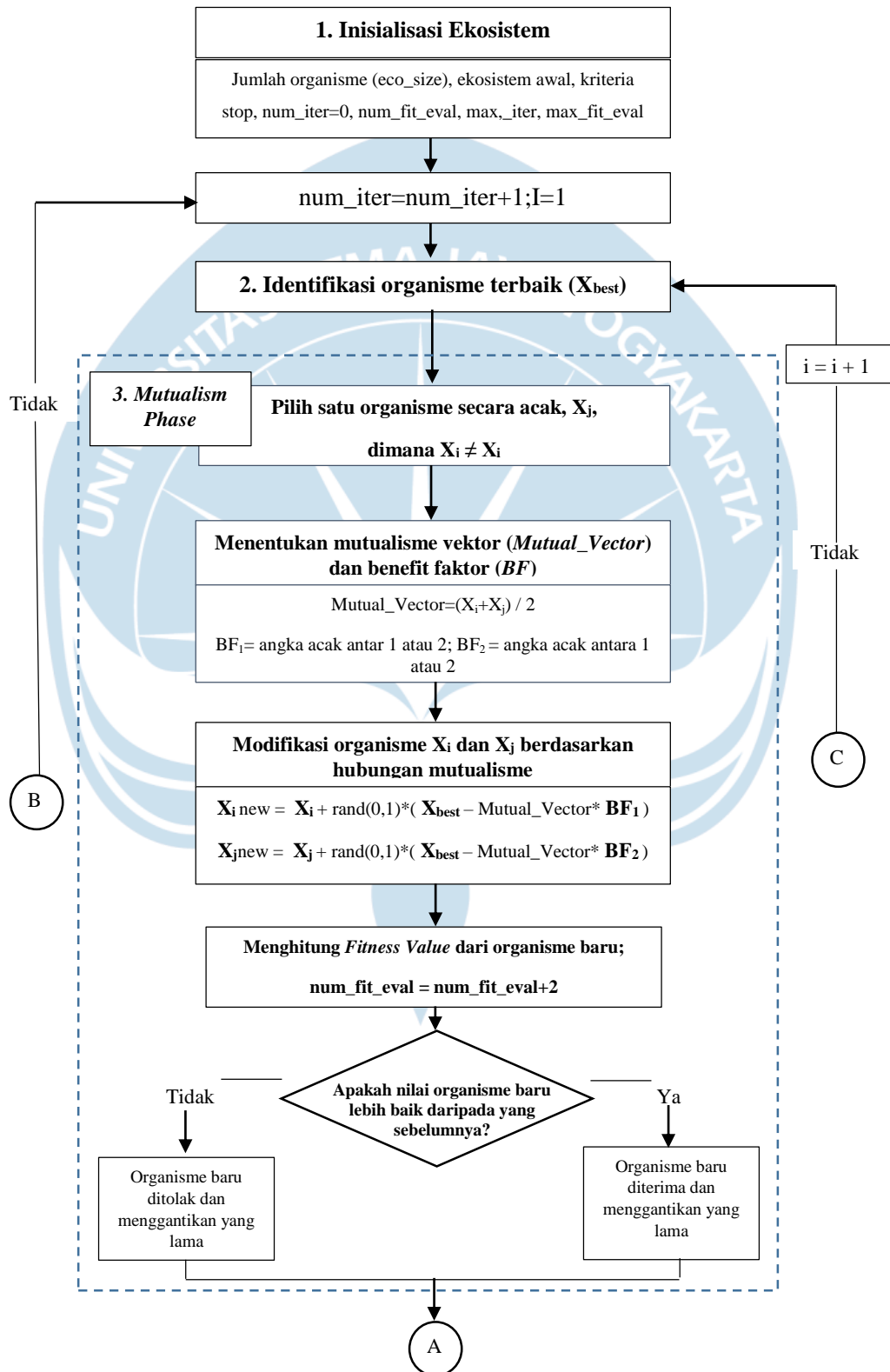
$$\text{Parasite_vector} = F * X_i + (1-F) * (\text{rand}(0,1) * (ub - lb) + lb) \dots \dots \dots (2.40)$$

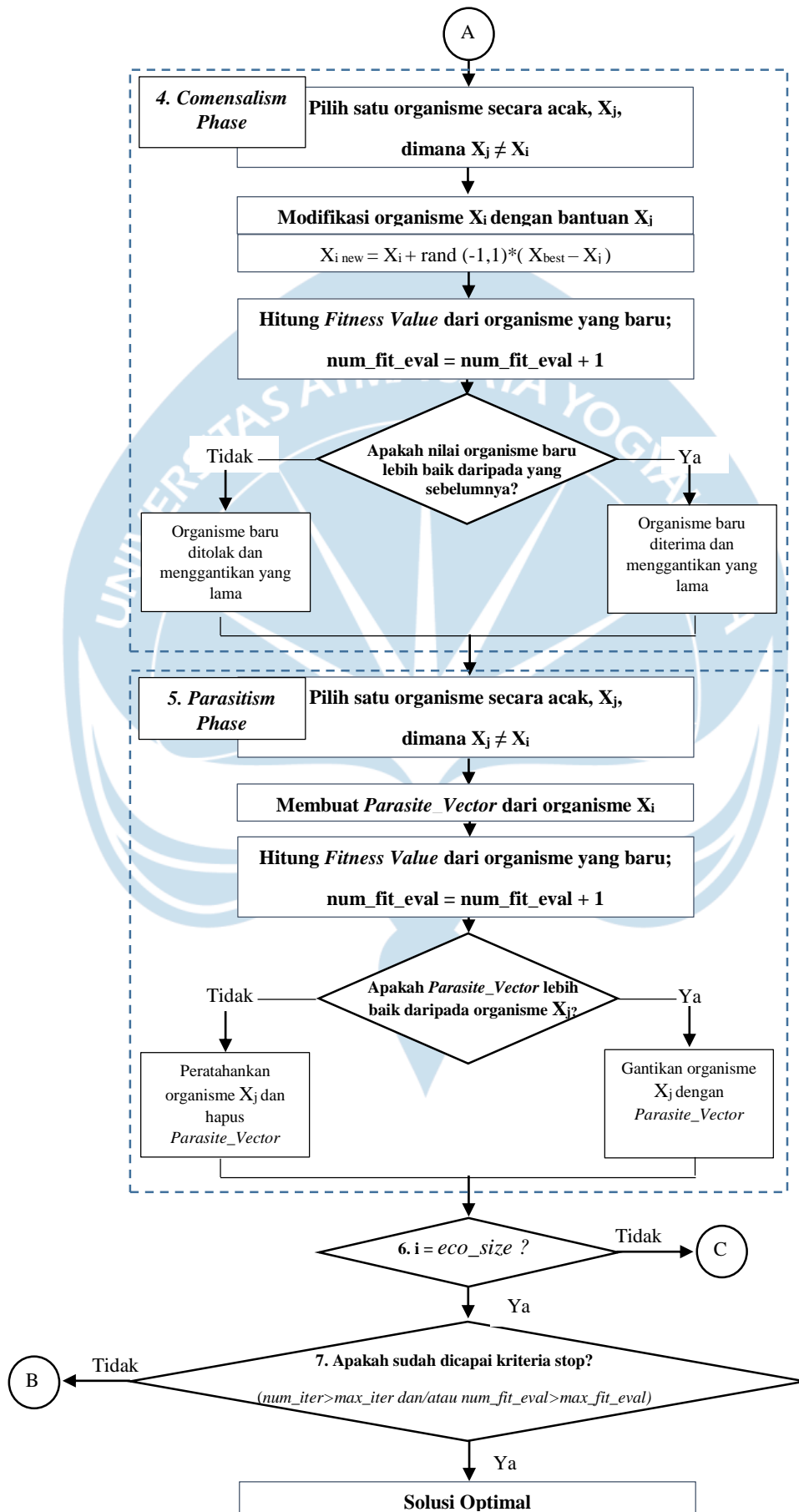
dengan F merupakan matriks biner acak yang bertujuan untuk menentukan secara acak variabel-variabel solusi yang akan dimodifikasi ; ub dan lb menunjukkan batas atas dan bawah dari rentang pencarian variabel solusi.

Organisme X_i dipilih secara acak dari ekosistem dan berfungsi sebagai *host* atau induk dari parasit tersebut. Setelah terjadi interaksi, *parasite_vector* akan menggantikan organisme X_j di ekosistem jika *parasite_vector* memiliki nilai objektif yang lebih baik. Tetapi jika nilai objektif X_j lebih baik, maka X_i akan tetap berada di ekosistem tersebut.

Adapun kode program dari algoritma SOS ini dapat dilihat secara terbuka melalui <http://cice.petra.ac.id/wordpress/wp-content/uploads/2017/06/SOS.rar>.

4. Bagan Alir Algoritma SOS





Gambar 2.3 Bagan Alir Algoritma SOS

H. Algoritma Deteksi Kerusakan Struktur

Seiring dengan berjalannya waktu, pengembangan metode deteksi kerusakan struktur tidak hanya sampai pada tingkat 2, yaitu untuk menentukan lokasi elemen yang mengalami kerusakan. Beberapa peneliti melakukan analisis sampai pada deteksi kerusakan struktur tingkat 3, yaitu menentukan lokasi kerusakan serta memprediksi penurunan kekuatan/kekakuan struktur yang mengalami kerusakan.

Meruane, V. & Heylen (2008) melakukan penelitian deteksi kerusakan struktur menggunakan algoritma genetika paralel riil. Digunakan balok beton bertulang yang diberi beban statik non-simetris secara meningkat untuk menghasilkan *crack* pada balok tersebut. Kemudian kerusakan diprediksi dengan hanya menggunakan beberapa ragam dan frekuensi. Dari penelitian ini terlihat bahwa metode deteksi kerusakan yang digunakan menunjukkan hasil yang cepat dan efisien. Global minimum dari fungsi objektif dapat dicapai, meskipun terdapat beberapa eror pada pemodelan dan *noise* pada saat dilakukan uji eksperimen. Hasil juga menunjukkan algoritma genetika paralel memberikan peningkatan performa jika dibandingkan dengan algoritma genetika sekuensial. Meskipun semua elemen pada *finite element model* digunakan sebagai parameter yang diperbaharui sampai dengan kondisi paling ekstrim, kerusakan yang dideteksi memiliki kecocokan yang baik dengan pola kerusakan yang dihasilkan dari perhitungan secara eksperimen.

Penelitian yang dilakukan Saberi dan Kaveh (2015) menggunakan metode gaya sisa dan algoritma *charged system search (CSS Algorithm)* untuk mendeteksi kerusakan struktur sampai dengan tingkat 3 pada struktur ruang (*space structur*). Pada awalnya digunakan metode gaya sisa untuk mendeteksi elemen yang

mengalami kerusakan. Namun metode ini tidak dapat membedakan derajat kebebasan yang berhubungan dengan elemen yang mengalami kerusakan karena mengandung *noise*, maka digunakan digunakan vektor gaya sisa. Dengan menggunakan vektor ini, adanya *noise* dapat dihilangkan secara signifikan. Kemudian algoritma CSS diterapkan guna menentukan besarnya tingkat kerusakan struktur tersebut.

Frans dan Arfiadi (2017) melakukan penelitian tahap ini menggunakan teknik MAC-FMSO. Teknik ini adalah penggabungan dari modal assurance criterion (MAC) dan teknik Fast Multi Swarm Optimization (FMSO). Pada teknik ini terdapat 2 tahapan. Pertama mendapatkan lokasi struktur yang diidentifikasi mengalami kerusakan menggunakan teknik deteksi kerusakan struktur tingkat 2, seperti metode *mode shape curvature*, VBPLK, ataupun metode *strain energy*. Lalu pada tahap ke dua digunakan teknik MAC-FMSO untuk mendapatkan nilai penurunan kekakuan dari elemen struktur yang diidentifikasi rusak. Teknik ini diterapkan pada 3 jenis struktur, yaitu balok tumpuan sederhana dengan 3 skenario kerusakan, struktur rangka bidang 13 batang dengan 2 skenario kerusakan, dan yang terakhir struktur *shear building* dengan 3 skenario kerusakan. Berdasarkan hasil yang didapatkan, dapat disimpulkan bahwa teknik MAC-FMSO terbukti efektif dan efisien dalam mendeteksi kerusakan struktur sampai dengan tingkat 3. Teknik ini memberikan hasil yang akurat dalam memprediksi penurunan kekakuan struktur.

Kemudian Frans dan Arfiadi (2019) menggunakan *modal assurance criterion-firefly algorithm* (MAC-FA) untuk mendeteksi kerusakan struktur sampai dengan tingkat 3. Pada awalnya digunakan metode VBPLK untuk menentukan

lokasi struktur yang mengalami kerusakan kemudian digunakan MAC-FA untuk memprediksi penurunan/kekakuan struktur. Dua struktur portal bidang digunakan dalam penelitian ini, yaitu struktur portal bidang dengan 3 batang dan struktur portal bidang dengan 28 batang. Terhadap dua struktur tersebut dilakukan beberapa skenario kerusakan struktur untuk selanjutnya dilakukan analisis menggunakan metode MAC-FA. Hasil analisis menunjukkan metode MAC-FA dapat memprediksi dengan akurat penurunan kekakuan/kekuatan elemen struktur yang diidentifikasi mengalami kerusakan. Maka metode MAC-FA ini dapat digolongkan sebagai metode deteksi kerusakan struktur tingkat ketiga.

Berdasarkan penelitian-penelitian yang telah dijabarkan sebelumnya, penulis tertarik untuk melakukan penelitian deteksi kerusakan struktur sampai dengan tingkat 3 menggunakan metode lain, yaitu *modal assurance criterion-symbiotic organism search algorithm* (MAC-SOS).