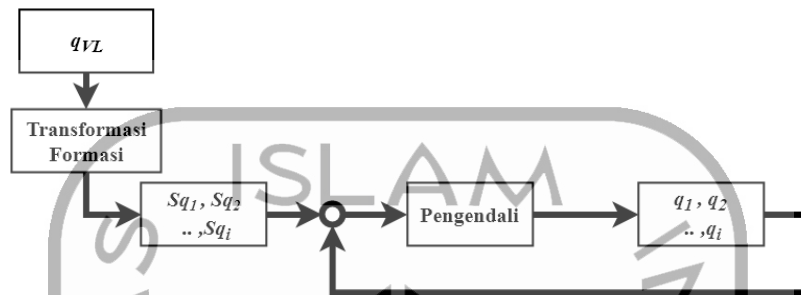


## BAB 3 METODOLOGI

### 3.1 Proses Pengendalian Formasi Robot Majemuk

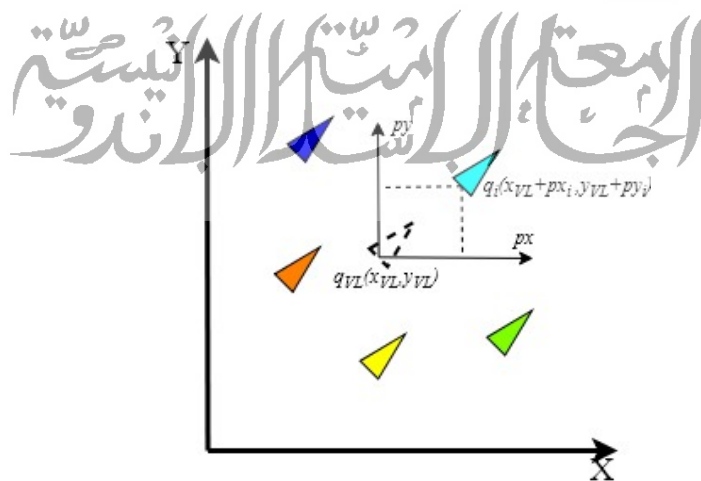
Proses pengendalian formasi yang digunakan pada robot majemuk tipe *unicycle* mengikuti diagram blok pada Gambar 3.1



Gambar 3.1: Diagram kendali formasi

Pengendalian dimulai dengan bentuk formasi yang dituju. Formasi dibentuk dengan *virtual leader* dengan model *unicycle* yang akan menjadi acuan untuk membuat formasi. Jika posisi *Virtual leader* ( $q_{VL}$ ) terhadap bidang adalah  $[x_{VL} \ y_{VL}]^T$  maka untuk membentuk formasi antar robot dituliskan pada Persamaan (3.1) .

$$Sq_i = \begin{bmatrix} px_i \\ py_i \end{bmatrix} + \begin{bmatrix} x_{VL} \\ y_{VL} \end{bmatrix} \quad (3.1)$$



Gambar 3.2: Formasi yang dibentuk berdasarkan posisi *virtual leader*

Pada Persamaan (3.1),  $Sq_i$  merupakan hasil transformasi  $q_i$  terhadap *virtual leader* dan yang akan menjadi target dari setiap robot  $q_i$  atau jika diilustrasikan bisa dilihat pada Gambar 3.2 yang menunjukkan formasi yang dibentuk berdasarkan posisi *virtual leader*.

Berdasarkan diagram blok Gambar 3.1, terdapat  $q_{VL}$  (*virtual leader*),  $Sq_i$  (target posisi robot  $i$ ), dan  $q_i$  (posisi robot- $i$  pada bidang) yang mengikuti persamaan (2.1). Namun untuk mendapatkan posisi terbaru maka persamaan (2.1) tersebut diintegrasikan dan didapat persamaan (3.2)

$$q_i(t+1) = \begin{bmatrix} x_i(t+1) \\ y_i(t+1) \\ \theta_i(t+1) \end{bmatrix} = \begin{bmatrix} \cos(\theta_i(t)) & 0 \\ \sin(\theta_i(t)) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_i(t) \\ \omega_i(t) \end{bmatrix} \times \Delta t + \begin{bmatrix} x_i(t) \\ y_i(t) \\ \theta_i(t) \end{bmatrix} \quad (3.2)$$

Namun yang perlu diketahui untuk menghitung persamaan (3.2) maka terdapat sinyal kendali yang akan dihitung untuk  $q_i$  (robot- $i$ ), yakni  $[v_i \ \omega_i]^T$  sebagai sinyal kendali dan  $\Delta t$  sebagai *sampling time* atau selisih antar waktu. Sinyal kendali tersebut dihitung dengan dua metode yang telah diajukan yakni pengendali distribusi berbasis konsensus, dan *vector field*.

Untuk melakukan pengendalian formasi dengan metode yang diajukan, maka diperlukan penyesuaian. Pada metode *vector field* proses pengendaliannya menggunakan posisi target yang telah dibentuk sebelumnya berdasarkan posisi *virtual leader*. Dan posisi target yang ditentukan pada Persamaan (3.2), yang dimana  $[x_{it} \ y_{it}]$  disubstitusikan ke beberapa persamaan (2.2) – (2.15).

$$q_{VL,i} = \begin{bmatrix} x_{it} \\ y_{it} \end{bmatrix} = \begin{bmatrix} px_i \\ py_i \end{bmatrix} + \begin{bmatrix} x_{VL} \\ y_{VL} \end{bmatrix} \quad (3.3)$$

Kemudian untuk metode pengendalian terdistribusi berbasis konsensus pada Persamaan (2.18) – (2.22) diperlukan penyesuaian pada transformasi *virtual leader* atau  $z_{10}$ ,  $z_{20}$ , dan  $z_{30}$ . Jika pada  $z_{10}$ ,  $z_{20}$ , dan  $z_{30}$  bergantung dari posisi dan masukan *virtual leader* maka Persamaan (2.18) di modifikasi menjadi Persamaan (3.4) yang akan digunakan pada robot- $i$ . Sedangkan pada robot- $i$  tetap menggunakan persamaan yang sama dengan sebelumnya.

$$z_{10}(t) = \theta_{VL}(t),$$

$$z_{20}(t) = x_{VL}(t) \cos \theta_{VL}(t) + y_{VL}(t) \sin \theta_{VL}(t) + k_0 \text{sign}(\omega_{VL}(t)) z_{30}(t), \quad (3.4)$$

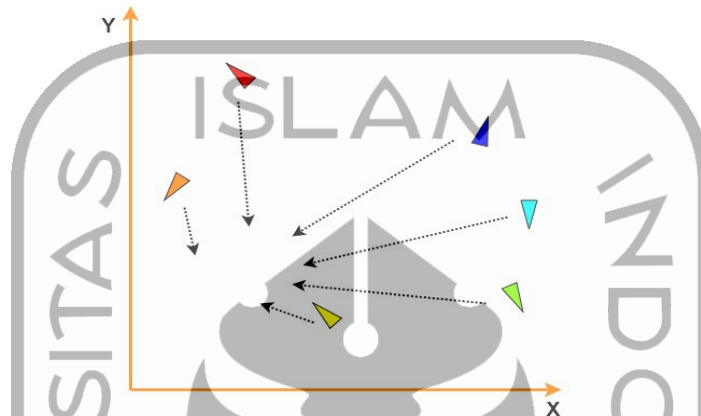
$$z_{30}(t) = x_{VL}(t) \sin \theta_{VL}(t) + y_{VL}(t) \cos \theta_{VL}(t),$$

### 3.2 Perancangan Simulasi dan Metode Analisis

#### 3.2.1 Skenario Simulasi

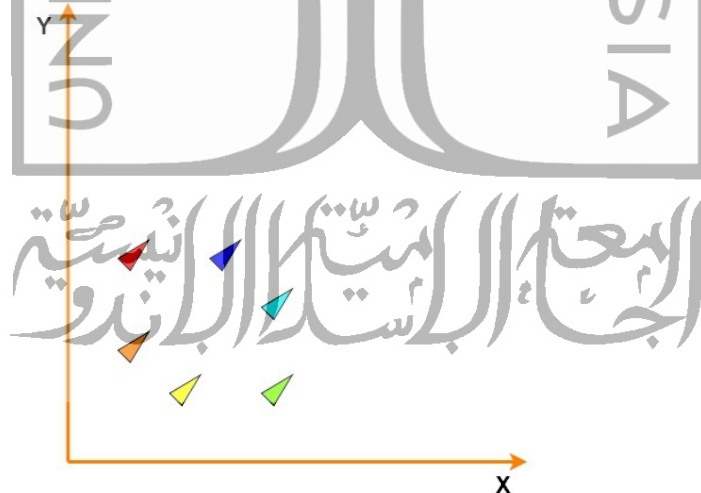
Pada penelitian tentang pengendalian formasi, peneliti memerlukan skenario simulasi untuk menjadi acuan untuk pengendalian formasi ini. Skenario simulasi adalah sebagai berikut

- Beberapa model robot *unicycle* diletakkan secara acak pada ruang yang tersedia, yang kemudian akan berkumpul



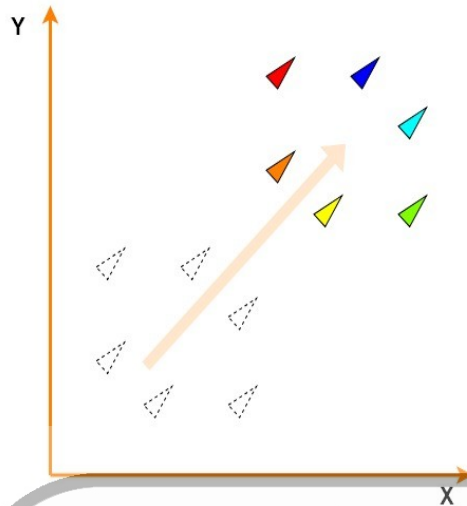
Gambar 3.3: Model Robot unicycle diletakkan acak

- Kemudian robot akan terkumpul dengan formasi yang diinginkan,



Gambar 3.4: Robot model *unicycle* membentuk formasi

- Setelah Robot membentuk formasi, maka diarahkan robot tersebut berjalan dengan mempertahankan formasinya. Lintasan yang dibentuk pada penelitian ini adalah lintasan lingkaran.



Gambar 3.5: Robot Model Unicycle akan berjalan sesuai track dan mempertahankan formasinya

Lintasan lingkaran dibentuk dengan menggunakan mengatur input  $v_{VL}$  dan  $\omega_{VL}$  pada *virtual leader* yang akan menjadi referensi setiap  $q_i$ .

### 3.3 Proses Analisis

Proses analisis dilakukan setelah simulasi dengan menggunakan dua metode pengendalian yang telah diuraikan pada subbab 3.1.2, yang kemudian mengakuisisi data simulasi untuk melakukan proses analisis. Proses analisis berupa perbandingan *error* pada robot satu dengan lainnya, perbandingan *error* formasi antar robot dan juga sinyal kendali ( $v_j$ ,  $\omega_j$ ) pada setiap robot. Target kesesuaian formasi ataupun mencapai target pada penelitian ini hanya dibatasi jika tingkat *error* kurang dari 5 cm pada *error* formasi ataupun *error* posisi setiap robot.

Perbandingan *error* tersebut akan menghasilkan ketepatan setiap robot terhadap posisi yang diinginkan sesuai dengan Persamaan 3.23 – 3.25

$$e_x^i(t) = x_{it}(t) - x_i(t) \quad (3.23)$$

$$e_y^i(t) = y_{it}(t) - y_i(t) \quad (3.24)$$

Rumusan kesesuaian formasi dimulai dengan robot- $k$  yang memiliki transformasi formasi  $[px_i \ py_i]^T = [0 \ 0]$  seperti pada Persamaan (3.1) sebagai acuan untuk menghitung *error* formasi antar robot. Kemudian untuk mendapatkan *error* formasi dituliskan pada Persamaan (3.26) – (3.27)

$$e_{px}^i(t) = x_i(t) - (x_k(t) + px_i) \quad (3.26)$$

$$e_{py}^i(t) = y_i(t) - (y_k(t) + py_i) \quad (3.27)$$