Chapter 9

Localization in Arbitrary Dimensional Space

In this chapter, we extend the distributed localization problem of multi-agent systems in Chapter 7 from two-dimensional space to arbitrary dimensional space. This extension is practically useful because many applications of localization using (wireless) sensor networks are not limited to 2D space. For example, air quality monitoring and underwater information collection are instances in 3D space.

To solve localization in arbitrary dimensions, we develop an approach based on signed Laplacian matrices (as in Chapter 8 for arbitrary dimensional affine formation control). Note that the approach for solving localization in Chapter 7 based on complex Laplacian matrices was limited to 2D space, and cannot be used for higher dimensional localization.

We nevertheless adopt the same distributed localization scheme introduced in Chapter 7. Namely we consider a sensor network composed of a minority of *anchor* nodes that know their positions in the global coordinate frame (e.g. using a GPS), and the rest majority of *free* nodes that need to determine their global positions based on their local frames and locally sensed information (e.g. distances and bearing angles with respect to neighboring nodes).

Modeling the interacting sensor nodes by digraphs, we show that a necessary graphical condition to achieve d-dimensional localization ($d \geq 2$) is that the digraph contains a *spanning* $(d+1)$-*tree* whose $d+1$ roots are anchor nodes. This condition is the same as the one for achieving d-dimensional affine formation in Chapter 8. However, in the special case of $d = 2$, this condition differs from the one (i.e. spanning 2-tree) for achieving 2D localization in Chapter 7. This difference is due to distinct graphical requirements on designing appropriate entries for signed Laplacian matrices and for complex Laplacian matrices. Under the above graphical condition, we present a distributed algorithm to achieve localization in arbitrary dimensions.

9.1 Problem Formulation

Consider a network of n (> 1) agents that are stationary in d-dimensional space ($d \geq 2$), and a global coordinate frame Σ which is unknown to the agents. The agents labeled $1, \ldots, d+1$ (renumbering if necessary) are the *anchor agents*, whose positions $\xi_1, \ldots, \xi_{d+1} \in \mathbb{R}^d$ in Σ are known. The rest agents labeled $d+2, \ldots, n$ are the *free agents*, whose positions $\xi_{d+2}, \ldots, \xi_n \in \mathbb{R}^d$ in Σ are unknown and need to be determined by these individual free agents. Let

$$\xi_a := \begin{bmatrix} \xi_1 \\ \vdots \\ \xi_{d+1} \end{bmatrix} \in \mathbb{R}^{(d+1)d}, \quad \xi_f := \begin{bmatrix} \xi_{d+2} \\ \vdots \\ \xi_n \end{bmatrix} \in \mathbb{R}^{(n-d-1)d}$$

be the aggregated position vectors of the anchor and free agents, respectively. Write ξ in terms of ξ_a and ξ_f as follows:

$$\xi = \begin{bmatrix} \xi_a \\ \xi_f \end{bmatrix} \in \mathbb{R}^{nd}$$

and call ξ the *configuration* of the agents.

To determine its own position, each free agent i ($\in [d+2, n]$) is equipped with a *state* variable $x_i(k) \in \mathbb{R}^d$, which is a d-dimensional real vector and denotes the *estimate* of agent i's position ξ_i under the global frame Σ. The time $k \geq 0$ is a nonnegative integer and denotes the *discrete* time. Let

$$x_f(k) := \begin{bmatrix} x_{d+2}(k) \\ \vdots \\ x_n(k) \end{bmatrix} \in \mathbb{R}^{(n-d-1)d}$$

be the aggregated state vector of the free agents at time k. It is desired that

$$x_f(k) \to \xi_f \text{ as } k \to \infty.$$

For convenience, also let

$$x_a(k) := \begin{bmatrix} x_1(k) \\ \vdots \\ x_{d+1}(k) \end{bmatrix} \in \mathbb{R}^{(d+1)d}$$

be the aggregated state vector of the anchor agents, such that $x_a(k) = \xi_a$ for all $k \geq 0$ (i.e. the anchor agents know their positions in the global frame Σ from the initial time $k = 0$ and never update their estimates). Write $x(k) := [x_a(k)^\top \; x_f(k)^\top]^\top \in \mathbb{R}^{nd}$. Hence the aim of d-dimensional localization is to achieve

$$\lim_{k\to\infty} x(k) = \xi.$$

We model the interconnection structure of the networked agents by a digraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$: Each *node* in $\mathcal{V} = \{1, ..., n\}$ stands for an agent, and each directed *edge* (j, i) in $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ denotes that agent i can obtain the relative state information from agent j. The *neighbor set* of agent i is $\mathcal{N}_i := \{j \in \mathcal{V} \mid (j, i) \in \mathcal{E}\}$. For the $d+1$ anchor nodes (numbered $1, \ldots, d+1$ without loss of generality), since they do not update their states, even if they had neighbors, the corresponding incoming edges would be associated with weight 0. This is equivalent to considering that the anchor nodes do not have neighbors. For this reason, henceforth in this chapter we consider that $\mathcal{N}_i = \emptyset$ for all $i \in [1, d+1]$.

Moreover, consider that digraph $\mathcal{G}$ is weighted: each edge $(j, i) \in \mathcal{V}$ is associated with a real-valued weight $a_{ij} \in \mathbb{R}$. Hence the adjacency matrix $A = (a_{ij})$, degree matrix $D = \mathrm{diag}(A\mathbf{1})$, and Laplacian matrix $L = D - A$ are all real matrices. Note that the adjacency matrix A is not a nonnegative matrix in general; thus L is a *signed Laplacian matrix*. Since $\mathcal{N}_i = \emptyset$ for the anchor nodes $i \in [1, d+1]$, the signed Laplacian matrix L has the following structure:

$$L = \begin{bmatrix} L_{aa} & L_{af} \\ L_{fa} & L_{ff} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ L_{fa} & L_{ff} \end{bmatrix}. \tag{9.1}$$

Here $L_{fa} \in \mathbb{R}^{(n-d-1)\times(d+1)}$ and $L_{ff} \in \mathbb{R}^{(n-d-1)\times(n-d-1)}$.

To achieve localization in d dimensions, consider the distributed control

$$u_i(k) = \sum_{j\in\mathcal{N}_i} w_{ij}(x_j(k) - x_i(k)), \quad i \in [1, n]. \tag{9.2}$$

Here the control gain w_{ij} satisfies

$$\text{(i)} \sum_{j\in\mathcal{N}_i} w_{ij}(\xi_j - \xi_i) = 0 \tag{9.3}$$

$$\text{(ii)}\ w_{ij} = \epsilon_i a_{ij}, \quad \epsilon_i \in \mathbb{R} \setminus \{0\}. \tag{9.4}$$

This control u_i in (9.2) is in the same form as that for the 2D localization in Chapter 7: the gains w_{ij} are not simply the edge weights $a_{ij} \in \mathbb{R}$, but are real (nonzero) multiples of a_{ij} (9.4) and satisfy linear constraints with respect to the configuration ξ (9.3). In contrast with Chapter 7, here the gains w_{ij} are real numbers rather than complex ones.

Moreover, substituting (9.4) into (9.3) and removing the common multiple ϵ_i yield

$$\sum_{j \in \mathcal{N}_i} a_{ij}(\xi_j - \xi_i) = 0. \tag{9.5}$$

This in vector form is $(L \otimes I_d)\xi = 0$. In view of (9.1) we have

$$\begin{bmatrix} 0 & 0 \\ L_{fa} \otimes I_d & L_{ff} \otimes I_d \end{bmatrix} \begin{bmatrix} \xi_a \\ \xi_f \end{bmatrix} = 0$$

and thereby the following holds:

$$(L_{ff} \otimes I_d)\xi_f = -(L_{fa} \otimes I_d)\xi_a. \tag{9.6}$$

The above equation relates the configuration of the free agents to that of the anchor agents through appropriate multiplications of submatrices of the signed Laplacian matrix.

Arbitrary Dimensional Localization Problem:

Consider a network of agents (stationary in a d-dimensional space) interconnected through a digraph and a configuration $\xi := [\xi_a^\top \ \xi_f^\top]^\top \in \mathbb{R}^{nd}$, which represents the fixed positions of the agents under the global coordinate frame Σ. Here $\xi_a \in \mathbb{R}^{(d+1)d}$ is known but $\xi_f \in \mathbb{R}^{(n-d-1)d}$ is unknown. Design a distributed algorithm using the control u_i in (9.2) such that

$$\text{(i) } \operatorname{rank}(L) = n - d - 1$$
$$\text{(ii) } (\forall x_f(0) \in \mathbb{R}^{(n-d-1)d}) \lim_{k \to \infty} x_f(k) = \xi_f.$$

The first requirement (i) implies $\operatorname{rank}(L_{ff}) = n - d - 1$; namely L_{ff} is invertible. This means that $(L_{ff} \otimes I_d)$ is also invertible. Thus it follows from (9.6) that $\xi_f = -(L_{ff} \otimes I_d)^{-1}(L_{fa} \otimes I_d)\xi_a$. Therefore the second requirement (ii) becomes:

$$(\forall x_f(0) \in \mathbb{R}^{(n-d-1)d}) \lim_{k \to \infty} x_f(k) = -(L_{ff} \otimes I_d)^{-1}(L_{fa} \otimes I_d)\xi_a.$$

Example 9.1 *We provide an example to illustrate the localization problem in $d(=3)$ dimensions. As displayed in Fig. 9.1, eight agents are interconnected through a digraph; agents 1,2,3,4 are anchor agents while the rest five are free nodes. The neighbor sets of the agents are $\mathcal{N}_1 = \mathcal{N}_2 = \mathcal{N}_3 = \mathcal{N}_4 = \emptyset$, $\mathcal{N}_5 = \{1, 2, 6, 7\}$, $\mathcal{N}_6 = \{3, 4, 7, 8\}$, $\mathcal{N}_7 = \{1, 5, 6, 8\}$, and $\mathcal{N}_8 = \{4, 5, 6, 7\}$.*

Let the configuration $\xi = [\xi_1^\top \ \cdots \ \xi_8^\top]$ of the agents be the vector of eight (three-dimensional)

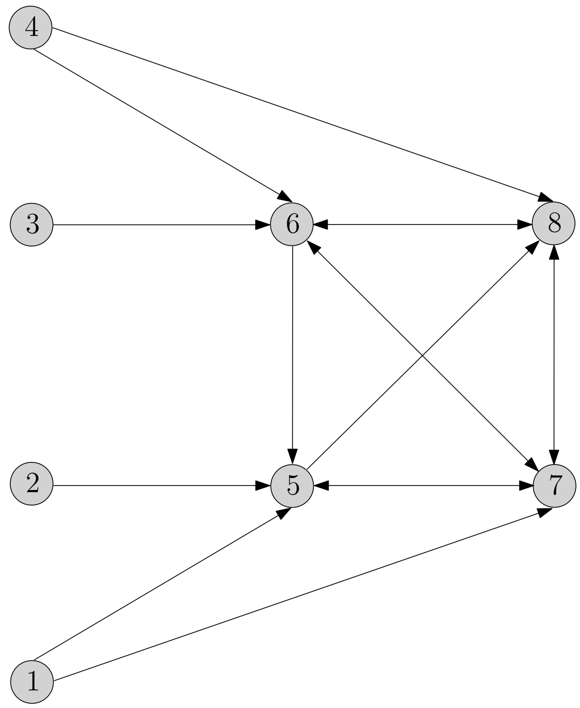

Figure 9.1: Illustrating example of eight agents

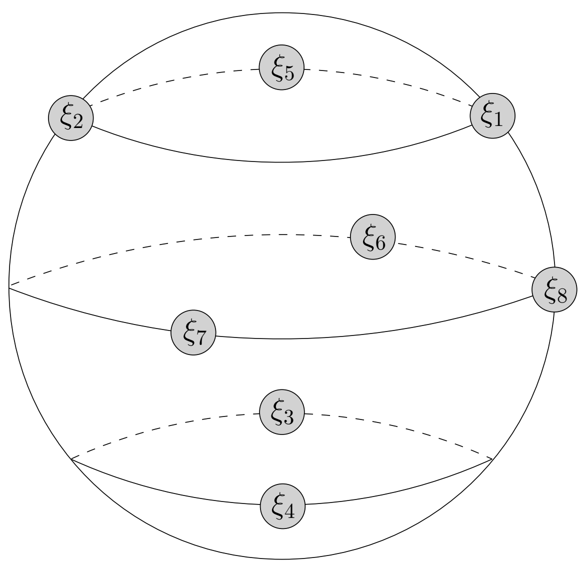

Figure 9.2: Illustrating example of a configuration of eight 3D points on unit sphere

points on the unit sphere (see Fig. 9.2), where

$$\xi_1 = \begin{bmatrix} \cos\frac{\pi}{4} \\ 0 \\ \sin\frac{\pi}{4} \end{bmatrix}, \xi_2 = \begin{bmatrix} -\cos\frac{\pi}{4} \\ 0 \\ \sin\frac{\pi}{4} \end{bmatrix}, \xi_3 = \begin{bmatrix} 0 \\ -\cos\frac{\pi}{4} \\ -\sin\frac{\pi}{4} \end{bmatrix}, \xi_4 = \begin{bmatrix} 0 \\ \cos\frac{\pi}{4} \\ -\sin\frac{\pi}{4} \end{bmatrix},$$
$$\xi_5 = \begin{bmatrix} 0 \\ -\cos\frac{\pi}{4} \\ \sin\frac{\pi}{4} \end{bmatrix}, \xi_6 = \begin{bmatrix} \cos\frac{\pi}{3} \\ -\sin\frac{\pi}{3} \\ 0 \end{bmatrix}, \xi_7 = \begin{bmatrix} -\cos\frac{\pi}{3} \\ \sin\frac{\pi}{3} \\ 0 \end{bmatrix}, \xi_8 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}.$$

The position vector of the anchor agents $\xi_a = [\xi_1^\top \ \xi_2^\top \ \xi_3^\top \ \xi_4^\top]^\top$ is known, while that of the free agents $\xi_f = [\xi_5^\top \ \xi_6^\top \ \xi_7^\top \ \xi_8^\top]^\top$ is unknown and needs to determined.
The localization problem in 3D is to design a distributed algorithm using the control u_i in (9.2) such that the rank of the signed Laplacian matrix L is $n-4$, and moreover the free agents' state vector asymptotically converges to ξ_f.

A necessary graphical condition for solving the d-dimensional localization problem is given below.

Proposition 9.1 *Suppose that there exists a distributed control u_i in (9.2) that solves the d-dimensional localization problem. Then the digraph contains a spanning $(d+1)$-tree whose $d+1$ roots are the $d+1$ anchor agents.*

Proof. Suppose that there exists a distributed control in (9.2) that solves the d-dimensional localization problem, but that the digraph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ does *not* contain a spanning $(d+1)$-tree whose $d+1$ roots are the $d+1$ anchor agents. We will derive a contradiction that $\text{rank}(L) < n-d-1$, thereby proving that after all $\mathcal{G}$ must contain a spanning $(d+1)$-tree whose $d+1$ roots are the $d+1$ anchor agents.

There are two cases that need to be considered separately. First, the digraph contains a spanning $(d+1)$-tree but at least one of the $d+1$ roots is a free agent. In this case, the subdigraph of free agents contains at least a spanning tree (and at most a spanning $(d+1)$-tree). Hence $\text{rank}(L_{ff}) < n-d-1$. Since the anchor agents do not have neighbors, $\text{rank}(L) < n-d-1$.

The second case is that the digraph does not contain a spanning $(d+1)$-tree. Then it follows similarly to the proof of Proposition 8.1 that $\text{rank}(L) < n-d-1$.

Therefore in both cases above, a contradiction is derived to the solvability of the d-dimensional localization problem. The proof is now complete. □

Owing to Proposition 9.1, we shall henceforth assume the following graphical condition.

Assumption 9.1 *The digraph $\mathcal{G}$ modeling the interconnection structure of the networked agents contains a spanning $(d+1)$-tree whose $d+1$ roots are the $d+1$ anchor agents.*

Even if Assumption 9.1 holds, not every configuration $\xi \in \mathbb{R}^{nd}$ may be determined by a distributed control u_i in (9.2). Similar to Example 8.2, if ξ is not generic, it is possible that $\text{rank}(L) < n-d-1$ for all signed Laplacian matrices satisfying $(L \otimes I_d)\xi = 0$. This means that the d-dimensional localization problem is not solvable. For this reason, and also the fact that the set of all non-generic configurations has Lebesgue measure zero after all, we assume that the configuration ξ is generic.

Assumption 9.2 *The configuration* $\xi = [\xi_a^\top \ \xi_f^\top]^\top \in \mathbb{R}^{nd}$ *is generic.*

9.2 Distributed Algorithm

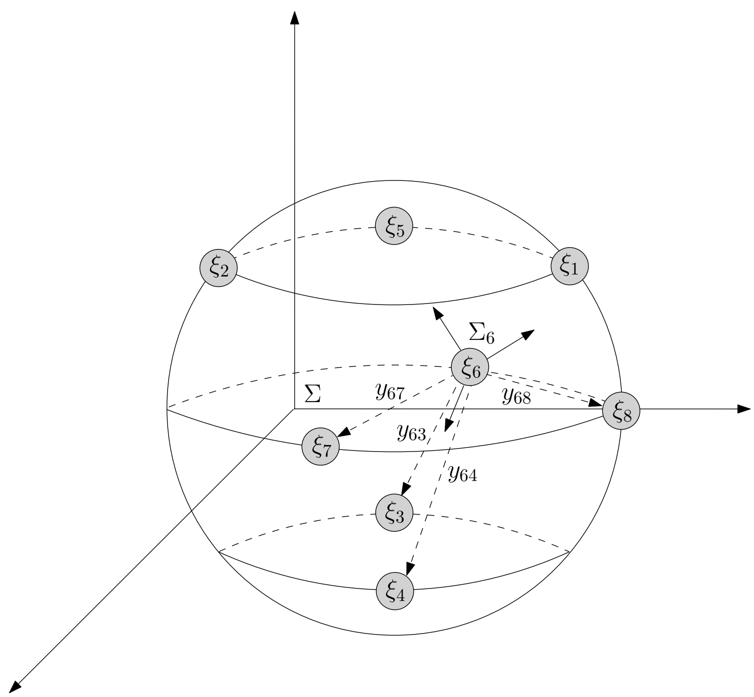

Figure 9.3: Illustration of design of real weights

Example 9.2 *Consider again Example 9.1, where the configuration* $\xi = [\xi_1^\top \ \cdots \ \xi_8^\top]^\top$ *of* *the agents consists of eight (three-dimensional) points on the unit sphere:*

$$\xi_1 = \begin{bmatrix} \cos\frac{\pi}{4} \\ 0 \\ \sin\frac{\pi}{4} \end{bmatrix}, \xi_2 = \begin{bmatrix} -\cos\frac{\pi}{4} \\ 0 \\ \sin\frac{\pi}{4} \end{bmatrix}, \xi_3 = \begin{bmatrix} 0 \\ -\cos\frac{\pi}{4} \\ -\sin\frac{\pi}{4} \end{bmatrix}, \xi_4 = \begin{bmatrix} 0 \\ \cos\frac{\pi}{4} \\ -\sin\frac{\pi}{4} \end{bmatrix},$$
$$\xi_5 = \begin{bmatrix} 0 \\ -\cos\frac{\pi}{4} \\ \sin\frac{\pi}{4} \end{bmatrix}, \xi_6 = \begin{bmatrix} \cos\frac{\pi}{3} \\ -\sin\frac{\pi}{3} \\ 0 \end{bmatrix}, \xi_7 = \begin{bmatrix} -\cos\frac{\pi}{3} \\ \sin\frac{\pi}{3} \\ 0 \end{bmatrix}, \xi_8 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}.$$

This configuration ξ is generic.
The anchor agents' configuration $\xi_a = [\xi_1^\top \ \xi_2^\top \ \xi_3^\top \ \xi_4^\top]^\top$ is known, and the free agents' configuration $\xi_f = [\xi_5^\top \ \xi_6^\top \ \xi_7^\top \ \xi_8^\top]^\top$ is to be determined. To this end, we consider using the simplest form of distributed control (9.2) by setting all $\epsilon_i = 1$:

$$x_i(k+1) = x_i(k) + \sum_{j\in\mathcal{N}_i} a_{ij}(x_j(k) - x_i(k)), \quad i \in [1, 8] \tag{9.7}$$

where $a_{ij} \in \mathbb{R}$ are real weights of edges (j, i) to be designed to satisfy (9.5):

$$\sum_{j\in\mathcal{N}_i} a_{ij}(\xi_j - \xi_i) = 0, \quad i \in [1, 8].$$

In the following we illustrate how the real weights may be designed locally to satisfy the above linear constraints. Each free agent $i \in [5, 8]$ has a local coordinate frame Σ_i, whose origin is the (stationary) position of agent i. The orientation of Σ_i is fixed, but the three offset angles $\alpha_i, \beta_i, \gamma_i$ (counterclockwise) with respect to the global coordinate frame Σ are unknown. These offset angles give rise to a (fixed) rotation matrix R_i relating the local frame Σ_i to the global Σ. For each neighbor (free or anchor) $j \in \mathcal{N}_i$, we assume that agent i can measure the relative position y_{ij} in Σ_i as

$$y_{ij} := R_i(\xi_j - \xi_i). \tag{9.8}$$

Since R_i is unknown, even though the relative position y_{ij} in Σ_i is known, $\xi_j - \xi_i$ in Σ is unknown. Substituting $\xi_j - \xi_i = R_i^{-1} y_{ij}$ into (9.5) and multiplying R_i from the left, we derive

$$\sum_{j\in\mathcal{N}_i} a_{ij} y_{ij} = 0. \tag{9.9}$$

Hence the weights a_{ij} may be designed based on the relative position y_{ij} under the local coordinate frame Σ_i.

For example, Fig. 9.3 provides an illustrative example. For agent 6, it has four neighbors $3, 4, 7, 8$. Thus we must find weights $a_{63}, a_{64}, a_{67}, a_{68}$ such that

$$a_{63}y_{63} + a_{64}y_{64} + a_{67}y_{67} + a_{68}y_{68} = 0.$$

The relative positions measured by agent 6 in its local frame Σ_6 are

$$y_{63} = \begin{bmatrix} 0 \\ -\cos\frac{\pi}{4} \\ \sin\frac{\pi}{4} \end{bmatrix}, y_{64} = \begin{bmatrix} \cos\frac{\pi}{3} \\ -\sin\frac{\pi}{3} \\ 0 \end{bmatrix}, y_{67} = \begin{bmatrix} -\cos\frac{\pi}{3} \\ \sin\frac{\pi}{3} \\ 0 \end{bmatrix}, y_{68} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}.$$

The local frame Σ_6 has (fixed) offset angles from the global Σ: $\alpha_6 = \frac{\pi}{4}$, $\beta_6 = \frac{\pi}{6}$, and $\gamma_6 = \frac{\pi}{3}$ (all counterclockwise with respect to Σ). Then the corresponding rotation matrix is

$$R_6 = \begin{bmatrix} \cos(\frac{\pi}{3}) & -\sin(\frac{\pi}{3}) & 0 \\ \sin(\frac{\pi}{3}) & \cos(\frac{\pi}{3}) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\frac{\pi}{6}) & 0 & \sin(\frac{\pi}{6}) & 0 \\ 0 & 1 & 0 \\ -\sin(\frac{\pi}{6}) & 0 & \cos(\frac{\pi}{6}) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\frac{\pi}{4}) & -\sin(\frac{\pi}{4}) \\ 0 & \sin(\frac{\pi}{4}) & \cos(\frac{\pi}{4}) \end{bmatrix}.$$

It is verified that

$$y_{6j} = R_6(\xi_j - \xi_6), \quad j = 3, 4, 6, 7.$$

Substituting the relative positions $y_{63}, y_{64}, y_{67}, y_{68}$ into the equation $a_{63}y_{63} + a_{64}y_{64} + a_{67}y_{67} + a_{68}y_{68} = 0$ yields

$$a_{63}\begin{bmatrix} -0.8437 \\ -0.2367 \\ -0.0857 \end{bmatrix} + a_{64}\begin{bmatrix} -1.4598 \\ 0.6964 \\ 0.7803 \end{bmatrix} + a_{67}\begin{bmatrix} -1.1875 \\ 0.3927 \\ 1.5607 \end{bmatrix} + a_{68}\begin{bmatrix} -0.1607 \\ 0.9464 \\ 0.2803 \end{bmatrix} = 0.$$

The above is a system of linear equations, with four unknowns (the weights) and three equations. Thus there are infinitely many solutions (indeed the solution space is one-dimensional). One solution is $a_{63} = -1, a_{64} = 1, a_{67} = -0.4082, a_{68} = -0.8165$.

Similarly we design other real weights to satisfy (9.9), and write (9.7) in vector form:

$x(k+1) = ((I - L) \otimes I_3)x(k)$ *where*

$$L = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -3.7321 & 0 & 0 & 4.7321 & -1.9319 & 1.9319 & 0 \\ 0 & 0 & -1 & 1 & 0 & 1.2247 & -0.4082 & -0.8165 \\ -1 & 0 & 0 & 0 & 1 & -0.9659 & -0.1494 & 1.1154 \\ 0 & 0 & 0 & -1 & -1 & 1 & 1 & 0 \end{bmatrix}.$$

It is verified that the signed Laplacian matrix L *has zero row sums and satisfies* $(L \otimes I_3)\xi = 0$. *Moreover, partition the matrix* L *according to anchor agents and free agents:*

$$L = \begin{bmatrix} L_{aa} & L_{af} \\ L_{fa} & L_{ff} \end{bmatrix}.$$

Thus $L_{aa} = L_{af} = 0$; $L_{fa} \in \mathbb{R}^{4\times 4}$ *and* $L_{ff} \in \mathbb{R}^{4\times 4}$. *It is checked that rank*$(L_{ff}) = 4$; *thus* L_{ff} *and* $(L_{ff} \otimes I_3)$ *are invertible. Therefore the first condition in the arbitrary dimensional localization problem is satisfied.*

It is left to verify the second condition that the state vector of the free agents $x_f(k)$ *converges to* $-(L_{ff} \otimes I_3)^{-1}(L_{fa} \otimes I_3)\xi_a$ *(when* $x_a(k) = \xi_a$ *for all* $k \geq 0$*). Fix* $\xi_a \in \mathbb{R}^{12}$. *First note that*

$$\bar{x} = \begin{bmatrix} \bar{x}_a \\ \bar{x}_f \end{bmatrix} = \begin{bmatrix} \xi_a \\ -(L_{ff} \otimes I_3)^{-1}(L_{fa} \otimes I_3)\xi_a \end{bmatrix}$$

is the unique fixed point of (9.7). To see this, substituting $\bar{x}$ *into (9.7) yields* $\bar{x}$, *which means that* $\bar{x}$ *is a fixed point of (9.7). Moreover, let*

$$\bar{x}' = \begin{bmatrix} \xi_a \\ \bar{x}'_f \end{bmatrix}$$

be another fixed point of (9.7), namely

$$\begin{bmatrix} \xi_a \\ \bar{x}'_f \end{bmatrix} = \left(\left(\begin{bmatrix} I_4 & 0 \\ 0 & I_4 \end{bmatrix} - \begin{bmatrix} 0 & 0 \\ L_{fa} & L_{ff} \end{bmatrix} \right) \otimes I_3 \right) \begin{bmatrix} \xi_a \\ \bar{x}'_f \end{bmatrix}$$
$$= \left(\left(\begin{bmatrix} I_4 & 0 \\ -L_{fa} & I_4 - L_{ff} \end{bmatrix} \right) \otimes I_3 \right) \begin{bmatrix} \xi_a \\ \bar{x}'_f \end{bmatrix}.$$

From the above we derive

$$\bar{x}'_f = -(L_{ff} \otimes I_3)^{-1}(L_{fa} \otimes I_3)\xi_a = \bar{x}_f.$$

This shows that $\bar{x}$ is the unique fixed point of (9.7), which in turn implies that starting from an arbitrary initial condition $x(0) = [\xi_a^\top \ x_f^\top(0)]^\top \in \mathbb{R}^{24}$, $x_f(k)$ converges to $-(L_{ff} \otimes I_3)^{-1}(L_{fa} \otimes I_3)\xi_a$ if and only if all the eigenvalues of $I_4 - L_{ff}$ lie inside the unit circle. Unfortunately, the eigenvalues of matrix $I_4 - L_{ff}$ are

$$-0.0967 + 0.2167\mathrm{j}, -0.0967 - 0.2167\mathrm{j}, 2.3807, -3.9946.$$

The last two eigenvalues lie outside the unit circle. Hence (9.7) is unstable and $x_f(k)$ diverges. To stabilize $x_f(k)$ to the desired fixed point $-(L_{ff} \otimes I_3)^{-1}(L_{fa} \otimes I_3)\xi_a$ (to satisfy the second requirement in the arbitrary dimensional localization problem), the unstable eigenvalues of $I_4 - L_{ff}$ must be moved inside the unit circle. This shows that simply setting all $\epsilon_i = 1$ in (9.2) does not work in general. In fact, ϵ_i need to be properly chosen in order to stabilize $I_4 - L_{ff}$.

Remark 9.1 *As illustrated in Example 9.2 for 3D localization, it is important for each free agent to have at least four neighbors to guarantee existence of (infinitely many) appropriate weights a_{ij} such that the signed Laplacian matrix L satisfies $(L \otimes I_3)\xi = 0$. If a free agent had only three or fewer neighbors, appropriate weights a_{ij} need not exist in general. This is why for solving general d-dimensional localization based on signed Laplacian matrices, the digraph must contain a spanning $(d+1)$-tree. Specializing to the case of $d = 2$, we need a digraph containing a spanning 3-tree for solving 2D localization based on signed Laplacian matrices. This graphical condition is stronger than the result of Chapter 7: there based on complex Laplacian matrices, 2D localization is solvable over a digraph containing a spanning 2-tree. Nevertheless, the signed Laplacian based approach can solve higher dimensional ($d \geq 3$) localization problem that cannot be dealt with by complex Laplacian matrices.*

In the following we describe a distributed algorithm using (9.2) in vector form, and will analyze its stability in relation to the values of ϵ_i in the next section.

Arbitrary Dimensional Localization Algorithm (ADLA):

Each anchor agent $i \in [1, \ldots, d+1]$ has a state variable $x_i(k) \in \mathbb{R}^d$ whose initial value is set to be $x_i(0) = \xi_i$ (which is known). Every free agent $i \in [d+2, \ldots, n]$ also has a state variable $x_i(k) \in \mathbb{R}^d$ whose initial value is an arbitrary d dimensional real vector (which is an estimate of the unknown ξ_i). Offline, each free agent i computes weights $a_{ij} \in \mathbb{R}$ based on the measured relative positions $y_{ij} = R_i(\xi_j - \xi_i)$ in (9.8) by solving

$$\sum_{j \in \mathcal{N}_i} a_{ij} y_{ij} = 0.$$

Then online, at each time $k \geq 0$, while each anchor agent stays put, i.e.

$$x_i(k+1) = x_i(k), \quad i \in [1, d+1]$$

each free agent i updates its $x_i(k)$ using the following local update protocol:

$$x_i(k+1) = x_i(k) + \epsilon_i \sum_{j \in \mathcal{N}_i} a_{ij}(x_j(k) - x_i(k)), \quad i \in [d+2, n] \tag{9.10}$$

where $\epsilon_i \in \mathbb{R} \setminus \{0\}$ is a (nonzero) real control gain.

Let $x := [x_1^\top \ \cdots \ x_n^\top]^\top \in \mathbb{R}^{nd}$ be the aggregated state vector of the networked agents, and

$$E = \operatorname{diag}(\epsilon_1, \ldots, \epsilon_n) \in \mathbb{R}^{n \times n}$$

the (diagonal and invertible) control gain matrix. Then the n equations (9.10) become

$$x(k+1) = ((I - EL) \otimes I_d)x(k). \tag{9.11}$$

Remark 9.2 *The above ADLA requires that the following information be available for each free agent $i \in [d+2, n]$:*

- y_{ij} *for all* $j \in \mathcal{N}_i$ *(offline computation of weights)*
- $x_j - x_i$ *for all* $j \in \mathcal{N}_i$ *(online state update).*

9.3 Convergence Result

The following is the main result of this section.

Theorem 9.1 *Suppose that Assumptions 9.1 and 9.2 hold. There exists a (diagonal and invertible) control gain matrix* $E = \text{diag}(\epsilon_1, \ldots, \epsilon_n)$ *such that ADLA solves the arbitrary dimensional localization problem.*

To prove Theorem 9.1, we analyze the eigenvalues of the matrix $(I - EL) \otimes I_d$ in (9.11). For this, the following fact is useful (which is the real counterpart of Lemma 7.1 and the discrete counterpart of Lemma 8.1).

Lemma 9.1 *Consider an arbitrary square real matrix* $M \in \mathbb{R}^{n \times n}$*. If all the principal minors of* M *are nonzero, then there exists an invertible diagonal matrix* $E = \text{diag}(\epsilon_1, \ldots, \epsilon_n) \in \mathbb{R}^{n \times n}$ *such that all the eigenvalues of* $I - EM$ *lie inside the unit circle.*

Proof: The proof is based on induction on n. For the base case $n = 1$, $M = m_{11}$ is a nonzero real scalar (as the principal minor of M is nonzero). Let $\epsilon_1 \in \mathbb{R}$ be such that $\epsilon_1 \in (0, \frac{1}{m_{11}})$. Then $EM = \epsilon_1 m_{11} \in (0, 1)$. Hence $1 - EM \in (0, 1)$, which lies inside the unit circle.

For the induction step, suppose that the conclusion holds for $M \in \mathbb{R}^{(n-1)\times(n-1)}$. Now consider $M \in \mathbb{R}^{n \times n}$, with all of its principal minors nonzero. Let M_1 be the submatrix of M with the last row and last column removed. Then all the principal minors of M_1 are nonzero, and by the hypothesis there exists an invertible diagonal matrix $E_1 = \text{diag}(\epsilon_1, \ldots, \epsilon_{n-1})$ such that all the eigenvalues $1 - \lambda_1, \ldots, 1 - \lambda_{n-1}$ of $I - E_1 M_1$ lie inside the unit circle. Now write

$$M = \begin{bmatrix} M_1 & M_2 \\ M_3 & m_{nn} \end{bmatrix}$$

where m_{nn} is a nonzero scalar (since all the principal minors of M are nonzero). Also let

$$E = \begin{bmatrix} E_1 & 0 \\ 0 & \epsilon_n \end{bmatrix}$$

for some real ϵ_n. Thus

$$I - EM = \begin{bmatrix} I & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} E_1 & 0 \\ 0 & \epsilon_n \end{bmatrix} \begin{bmatrix} M_1 & M_2 \\ M_3 & m_{nn} \end{bmatrix} = \begin{bmatrix} I - E_1 M_1 & -E_1 M_2 \\ -\epsilon_n M_3 & 1 - \epsilon_n m_{nn} \end{bmatrix}.$$

If $\epsilon_n = 0$, then

$$I - EM = \begin{bmatrix} I - E_1 M_1 & -E_1 M_2 \\ 0 & 1 \end{bmatrix}$$