Chapter 2
Semi-tensor Product of Matrices

2.1 Multiple-Dimensional Data

Roughly speaking, linear algebra mainly concerns two kinds of objects: vectors and matrices. An n-dimensional vector is expressed as $X = (x_1, x_2, \dots, x_n)$. Its elements are labeled by one index, i, where x_i is the ith element of X. For an $m \times n$ matrix

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & & & \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix},$$

elements are labeled by two indices, i and j, where $a_{i,j}$ is the element of A located in the ith row and jth column. In this way, it is easy to connect the dimension of a set of data with the number of indices. We define the dimension of a set of data as follows.

Definition 2.1 A set of data, labeled by k indices, is called a set of k-dimensional data. Precisely,

$$X = \{x_{i_1,i_2,\dots,i_k} \mid 1 \le i_j \le n_j, j = 1, 2, \dots, k\} \tag{2.1}$$

is a set of k-dimensional data. The cardinal number of X, denoted by $|X|$, is $|X| = n_1 n_2 \cdots n_k$.

In the following example we give an example of 3-dimensional data.

Example 2.1 Consider $\mathbb{R}^3$, with its canonical basis $\{e_1, e_2, e_3\}$. Any vector $X \in \mathbb{R}^3$ may then be expressed as $X = x_1 e_1 + x_2 e_2 + x_3 e_3$. When the basis is fixed, we simply use $X = (x_1, x_2, x_3)^{\mathrm{T}}$ to represent it. From simple vector algebra we know that in $\mathbb{R}^3$ there is a cross product, $\times$, such that for any two vectors $X, Y \in \mathbb{R}^3$ we

D. Cheng et al., *Analysis and Control of Boolean Networks*,
Communications and Control Engineering,
DOI 10.1007/978-0-85729-097-7_2, © Springer-Verlag London Limited 2011

have $X \times Y \in \mathbb{R}^3$, defined as follows:

$$X \times Y = \det\left(\begin{bmatrix} e_1 & e_2 & e_3 \\ x_1 & x_2 & x_3 \\ y_1 & y_2 & y_3 \end{bmatrix}\right). \tag{2.2}$$

Since the cross product is linear with respect to X as well as Y, it is a bilinear mapping. The value of the cross product is thus uniquely determined by its value on the basis. Write

$$e_i \times e_j = c_{ij}^1 e_1 + c_{ij}^2 e_2 + c_{ij}^3 e_3, \quad i, j = 1, 2, 3.$$

The coefficients form a set of 3-dimensional data,

$$\left\{c_{ij}^k \,\middle|\, i, j, k = 1, 2, 3\right\},$$

which are called the structure constants. Structure constants are easily computable. For instance,

$$e_1 \times e_2 = \det\left(\begin{bmatrix} e_1 & e_2 & e_3 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}\right) = e_3,$$

which means that $c_{12}^1 = c_{12}^2 = 0$, $c_{12}^3 = 1$. Similarly, we can determine all the structure constants:

$$\begin{array}{llllll}
c_{11}^1 = 0, & c_{11}^2 = 0, & c_{11}^3 = 0, & c_{12}^1 = 0, & c_{12}^2 = 0, & c_{12}^3 = 1, \\
c_{13}^1 = 0, & c_{13}^2 = -1, & c_{13}^3 = 0, & c_{21}^1 = 0, & c_{21}^2 = 0, & c_{21}^3 = -1, \\
c_{22}^1 = 0, & c_{22}^2 = 0, & c_{22}^3 = 0, & c_{23}^1 = 1, & c_{23}^2 = 0, & c_{23}^3 = 0, \\
c_{31}^1 = 0, & c_{31}^2 = 1, & c_{31}^3 = 0, & c_{32}^1 = -1, & c_{32}^2 = 0, & c_{32}^3 = 0, \\
c_{33}^1 = 0, & c_{33}^2 = 0, & c_{33}^3 = 0. & & &
\end{array}$$

Since the cross product is linear with respect to the coefficients of each vector, the structure constants uniquely determine the cross product. For instance, let $X = 3e_1 - e_3$ and $Y = 2e_2 + 3e_3$. Then

$$\begin{aligned}
X \times Y &= 6e_1 \times e_2 + 9e_1 \times e_3 - 2e_3 \times e_2 - 3e_3 \times e_3 \\
&= 6\left(c_{12}^1 e_1 + c_{12}^2 e_2 + c_{12}^3 e_3\right) + 9\left(c_{13}^1 e_1 + c_{13}^2 e_2 + c_{13}^3 e_3\right) \\
&\quad - 2\left(c_{32}^1 e_1 + c_{32}^2 e_2 + c_{32}^3 e_3\right) - 3\left(e_{33}^1 e_1 + c_{33}^2 e_2 + c_{33}^3 e_3\right) \\
&= 2e_1 - 9e_2 + 6e_3.
\end{aligned}$$

It is obvious that using structure constants to calculate the cross product in this way is very inconvenient, but the example shows that the cross product is uniquely determined by structure constants. So, in general, to define a multilinear mapping it is enough to give its structure constants.

Using structure constants to describe an algebraic structure is a powerful method.

Definition 2.2 [6] Let V be an n-dimensional vector space with coefficients in $\mathbb{R}$. If there is a mapping $* : V \times V \to V$, called the product of two vectors, satisfying

$$\begin{cases} (\alpha X + \beta Y) * Z = \alpha (X * Z) + \beta (Y * Z), \\ X * (\alpha Y + \beta Z) = \alpha (X * Y) + \beta (X * Z) \end{cases} \tag{2.3}$$

(where $\alpha, \beta \in \mathbb{R}$, $X, Y, Z \in V$), then $(V, *)$ is called an algebra.

Let $(V, *)$ be an algebra. If the product satisfies associative law, i.e.,

$$(X * Y) * Z = X * (Y * Z), \quad X, Y, Z \in V, \tag{2.4}$$

then it is called an associative algebra.

$\mathbb{R}^3$ with the cross product is obviously an algebra. It is also easy to check that it is not an associative algebra.

Let V be an n-dimensional vector space and $(V, *)$ an algebra. Choosing a basis $\{e_1, e_2, \dots, e_n\}$, the structure constants can be obtained as

$$e_i * e_j = \sum_{k=1}^{n} c_{ij}^k e_k, \quad i, j = 1, 2, \dots, n.$$

Although the structure constants $\{c_{ij}^k \mid i, j, k = 1, 2, \dots, n\}$ depend on the choice of basis, they uniquely determine the structure of the algebra. It is also easy to convert a set of structure constants, which correspond to a basis, to another set of structure constants, which correspond to another basis. For an algebra, the structure constants are always a set of 3-dimensional data.

Next, we consider an s-linear mapping on an n-dimensional vector space. Let V be an n-dimensional vector space and let $\phi : \underbrace{V \times V \times \cdots \times V}_{s} \to \mathbb{R}$, satisfying (for any $1 \le i \le s$, $\alpha, \beta \in \mathbb{R}$)

$$\begin{aligned} &\phi(X_1, X_2, \dots, \alpha X_i + \beta Y_i, \dots, X_{s-1}, X_s) \\ &\quad = \alpha \phi(X_1, X_2, \dots, X_i, \dots, X_{s-1}, X_s) + \beta \phi(X_1, X_2, \dots, Y_i, \dots, X_{s-1}, X_s). \end{aligned} \tag{2.5}$$

Equation (2.5) shows the linearity of ϕ with respect to each vector argument. Choosing a basis of V, $\{e_1, e_2, \dots, e_n\}$, the structure constants of ϕ are defined as

$$\phi(e_{i_1}, e_{i_2}, \dots, e_{i_s}) = c_{i_1, i_2, \dots, i_s}, \quad i_j = 1, 2, \dots, n, \ j = 1, 2, \dots, s.$$

Similarly, the structure constants, $\{c_{i_1, i_2, \dots, i_s} \mid i_1, \dots, i_s = 1, 2, \dots, n\}$, uniquely determine ϕ. Conventionally, ϕ is called a tensor, where s is called its covariant degree.

It is clear that for a tensor with covariant degree s, its structure constants form a set of s-dimensional data.

Example 2.2

1. In $\mathbb{R}^3$ we define a three linear mapping as

$$\phi(X, Y, Z) = \langle X \times Y, Z \rangle, \quad X, Y, Z \in \mathbb{R}^3,$$

   where $\langle \cdot, \cdot \rangle$ denotes the inner product. Its geometric interpretation is the volume of the parallelogram with X, Y, Z as three adjacent edges [when (X, Y, Z) form a right-hand system, the volume is positive, otherwise, the volume is negative]. It is obvious that ϕ is a tensor with covariant degree 3.
2. In $\mathbb{R}^3$ we can define a four linear mapping as

$$\psi(X, Y, Z, W) = \langle X \times Y, Z \times W \rangle, \quad X, Y, Z, W \in \mathbb{R}^3.$$

   Obviously, ψ is a tensor of covariant degree 4.

Next, we consider a more general case. Let $\mu : V \to \mathbb{R}$ be a linear mapping on V,

$$\mu(e_i) = c_i, \quad i = 1, \ldots, n.$$

Then, μ can be expressed as

$$\mu = c_1 e_1^* + c_2 e_2^* + \cdots + c_n e_n^*,$$

where $e_i^* : V \to \mathbb{R}$ satisfies

$$e_i^*(e_j) = \delta_{i,j} = \begin{cases} 1, & i = j, \\ 0, & i \neq j. \end{cases}$$

It can be seen easily that the set of linear mappings on V forms a vector space, called the dual space of V and denoted by V^*.

Let $X = x_1 e_1 + x_2 e_2 + \cdots + x_n e_n \in V$ and $\mu = \mu_1 e_1^* + \mu_2 e_2^* + \cdots + \mu_n e_n^* \in V^*$. When the basis and the dual basis are fixed, $X \in V$ can be expressed as a column vector and $\mu \in V^*$ can be expressed as a row vector, i.e.,

$$X = (a_1, a_2, \ldots, a_n)^{\mathrm{T}}, \qquad \mu = (c_1, c_2, \ldots, c_n).$$

Using these vector forms, the action of μ on X can be expressed as their matrix product:

$$\mu(X) = \mu X = \sum_{i=1}^{n} a_i c_i, \quad \mu \in V^*, X \in V.$$

Let $\phi : \underbrace{V^* \times \cdots \times V^*}_{t} \times \underbrace{V \times \cdots \times V}_{s} \to \mathbb{R}$ be an $(s + t)$-fold multilinear mapping. Then, ϕ is said to be a tensor on V with covariant degree s and contravariant degree t. Denote by $\mathscr{T}_t^s$ the set of tensors on V with covariant degree s and contravariant degree t.

If we define

$$c_{j_1,j_2,\dots,j_t}^{i_1,i_2,\dots,i_s} := \phi\big(e_{i_1}, e_{i_2}, \dots, e_{i_s}, e_{j_1}^*, e_{j_2}^*, \dots, e_{j_t}^*\big),$$

then

$$\big\{c_{j_1,j_2,\dots,j_t}^{i_1,i_2,\dots,i_s} \,\big|\, 1 \le i_1, \dots, i_s, j_1, \dots, j_t \le n\big\}$$

is the set of structure constants of ϕ. Structure constants of $\phi \in \mathscr{T}_t^s$ form a set of $(s+t)$-dimensional data.

Next, we consider how to arrange higher-dimensional data. In linear algebra one-dimensional data are arranged as a column or a row, called a vector, while two-dimensional data are arranged as a rectangle, called a matrix. In these forms matrix computation becomes a very convenient and powerful tool for dealing with one- or two-dimensional data. A question which then naturally arises is how to arrange three-dimensional data. A cubic matrix approach has been proposed for this purpose [1, 2] and has been used in some statistics problems [8–10], but, in general, has not been very successful. The problem is: (1) cubic matrices cannot be clearly expressed in a plane (i.e., on paper), (2) the conventional matrix product does not apply, hence some new product rules have to be produced, (3) it is very difficult to generalize this approach to even higher-dimensional cases.

The basic idea concerning the semi-tensor product of matrices is that no matter what the dimension of the data, they are arranged in one- or two-dimensional form. By then properly defining the product, the hierarchy structure of the data can be automatically determined. Hence the data arrangement is important for the semi-tensor product of data.

Definition 2.3 Suppose we are given a set of data S with $\prod_{i=1}^k n_i$ elements and, as in (2.1), the elements of x are labeled by k indices. Moreover, suppose the elements of x are arranged in a row (or a column). It is said that the data are labeled by indices $i_1, \dots, i_k$ according to an ordered multi-index, denoted by *Id* or, more precisely,

$$Id(i_1, \dots, i_k; n_1, \dots, n_k),$$

if the elements are labeled by $i_1, \dots, i_k$ and arranged as follows: Let i_t, $t = 1, \dots, k$, run from 1 to n_t with the order that $t = k$ first, then $t = k-1$, and so on, until $t = 1$. Hence, $x_{\alpha_1,\dots,\alpha_k}$ is ahead of $x_{\beta_1,\dots,\beta_k}$ if and only if there exists $1 \le j \le k$ such that

$$\alpha_i = \beta_i, \quad i = 1, \dots, j-1, \qquad \alpha_j < \beta_j.$$

If the numbers $n_1, \dots, n_k$ of $i_1, \dots, i_k$ are equal, we may use

$$Id(i_1, \dots, i_k; n) := Id(i_1, \dots, i_k; n, \dots, n).$$

If n_i are obviously known, the expression of *Id* can be simplified as

$$Id(i_1, \dots, i_k) := Id(i_1, \dots, i_k; n_1, \dots, n_k).$$

Example 2.3

1. Assume $x = \{x_{ijk} \mid i = 1,2,3; j = 1,2; k = 1,2\}$. If we arrange the data according to the ordered multi-index $Id(i, j, k)$, they are

$$x_{111}, x_{112}, x_{121}, x_{122}, x_{211}, x_{212}, x_{221}, x_{222}, x_{311}, x_{312}, x_{321}, x_{322}.$$

   If they are arranged by $Id(j, k, i)$, they become

$$x_{111}, x_{211}, x_{311}, x_{112}, x_{212}, x_{312}, x_{121}, x_{221}, x_{321}, x_{122}, x_{222}, x_{322}.$$

2. Let $x = \{x_1, x_2, \ldots, x_{24}\}$. If we use λ_1, λ_2, λ_3 to express the data in the form $a_i = a_{\lambda_1,\lambda_2,\lambda_3}$, then under different *Id*'s they have different arrangements:
   (a) Using the ordered multi-index $Id(\lambda_1, \lambda_2, \lambda_3; 2, 3, 4)$, the elements are arranged as

$$\begin{array}{cccc} x_{111} & x_{112} & x_{113} & x_{114} \\ x_{121} & x_{122} & x_{123} & x_{124} \\ x_{131} & x_{132} & x_{133} & x_{134} \\ \vdots & & & \\ x_{231} & x_{232} & x_{233} & x_{234}. \end{array}$$

   (b) Using the ordered multi-index $Id(\lambda_1, \lambda_2, \lambda_3; 3, 2, 4)$, the elements are arranged as

$$\begin{array}{cccc} x_{111} & x_{112} & x_{113} & x_{114} \\ x_{121} & x_{122} & x_{123} & x_{124} \\ x_{211} & x_{212} & x_{213} & x_{214} \\ \vdots & & & \\ x_{321} & x_{322} & x_{323} & x_{324}. \end{array}$$

   (c) Using the ordered multi-index $Id(\lambda_1, \lambda_2, \lambda_3; 4, 2, 3)$, the elements are arranged as

$$\begin{array}{ccc} x_{111} & x_{112} & x_{113} \\ x_{121} & x_{122} & x_{123} \\ x_{211} & x_{212} & x_{213} \\ \vdots & & \\ x_{421} & x_{422} & x_{423}. \end{array}$$

Note that in the above arrangements the data are divided into several rows, but this is simply because of spatial restrictions. Also, in this arrangement the hierarchy structure of the data is clear. In fact, the data should be arranged into one row.

Different *Id*'s, corresponding to certain index permutations, cause certain permutations of the data. For convenience, we now present a brief introduction to the permutation group. Denote by $\mathbf{S}_k$ the permutations of k elements, which form a group called the kth order permutation group. We use $1, \dots, k$ to denote the k elements. If we suppose that $k = 5$, then $\mathbf{S}_5$ consists of all possible permutations of five elements: $\{1, 2, 3, 4, 5\}$. An element $\sigma \in \mathbf{S}_5$ can be expressed as

$$\sigma = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 2 & 3 & 1 & 5 & 4 \end{bmatrix} \in \mathbf{S}_5.$$

That is, σ changes 1 to 2, 2 to 3, 3 to 1, 4 to 5, and 5 to 4. σ can also be simply expressed in a rotational form as

$$\sigma = (1, 2, 3)(4, 5).$$

Let $\mu \in \mathbf{S}_5$ and

$$\mu = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 4 & 3 & 2 & 1 & 5 \end{bmatrix}.$$

The product (group operation) on $\mathbf{S}_5$ is then defined as

$$\mu\sigma = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 2 & 3 & 1 & 5 & 4 \\ \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ 3 & 2 & 4 & 5 & 1 \end{bmatrix},$$

that is, $\mu\sigma = (1, 3, 4, 5)$.

If, in (2.1), the data x are arranged according to the ordered multi-index $Id(i_1, \dots, i_k)$, it is said that the data are arranged in a natural order. Of course, they may be arranged in the order of $(i_{\sigma(1)}, \dots, i_{\sigma(k)})$, that is, letting index $i_{\sigma(k)}$ run from 1 to $n_{\sigma(k)}$ first, then letting $i_{\sigma(k-1)}$ run from 1 to $n_{\sigma(k-1)}$, and so on. It is obvious that a different Id corresponds to a different data arrangement.

Definition 2.4 Let $\sigma \in \mathbf{S}_k$ and x be a set of data with $\prod_{i=1}^{k} n_i$ elements. Arrange x in a row or a column. It is said that x is arranged by the ordered multi-index $Id(i_{\sigma(1)}, \dots, i_{\sigma(k)}; n_{\sigma(1)}, \dots, n_{\sigma(k)})$ if the indices $i_1, \dots, i_k$ in the sequence are running in the following order: first, $i_{\sigma(k)}$ runs from 1 to $n_{\sigma(k)}$, then $i_{\sigma(k-1)}$ runs from 1 to $n_{\sigma(k-1)}$, and so on, until, finally, $i_{\sigma(1)}$ runs from 1 to $n_{\sigma(1)}$.

We now introduce some notation. Let $a \in \mathbb{Z}$ and $b \in \mathbb{Z}_+$. As in the programming language C, we use $a\%b$ to denote the remainder of a/b, which is always nonnegative, and $[t]$ for the largest integer that is less than or equal to t. For instance,

$$100\%3 = 1, \qquad 100\%7 = 2, \qquad (-7)\%3 = 2,$$
$$\left[\frac{7}{3}\right] = 2, \qquad [-1.25] = -2.$$

It is easy to see that

$$a = \left[\frac{a}{b}\right] b + a\%b. \tag{2.6}$$

Next, we consider the index-conversion problem. That is, we sometimes need to convert a single index into a multi-index, or vice versa. Particularly, when we need to deform a matrix into a designed form using computer, index conversion is necessary. The following conversion formulas can easily be proven by mathematical induction.

Proposition 2.1 *Let S be a set of data with $n = \prod_{i=1}^{k} n_i$ elements. The data are labeled by single index as $\{x_i\}$ and by k-fold index, by the ordered multi-index $Id(\lambda_1, \dots, \lambda_k; n_1, \dots, n_k)$, as*

$$S = \{s_p \mid p = 1, \dots, n\} = \{s_{\lambda_1,\dots,\lambda_k} \mid 1 \le \lambda_i \le n_i; i = 1, \dots, k\}.$$

We then have the following conversion formulas:

1. *Single index to multi-index. Defining $p_k := p - 1$, the single index p can be converted into the order of the ordered multi-index $Id(i_1, \dots, i_k; n_1, \dots, n_k)$ as $(\lambda_1, \dots, \lambda_k)$, where λ_i can be calculated recursively as*

$$\begin{cases} \lambda_k = p_k \% n_k + 1, \\ p_j = [\frac{p_{j+1}}{n_{j+1}}], \quad \lambda_j = p_j \% n_j + 1, \quad j = k-1, \dots, 1. \end{cases} \tag{2.7}$$

2. *Multi-index to single index. From multi-index $(\lambda_1, \dots, \lambda_k)$ in the order of $Id(i_1, \dots, i_k; n_1, \dots, n_k)$ back to the single index, we have*

$$p = \sum_{j=1}^{k-1} (\lambda_j - 1) n_{j+1} n_{j+2} \cdots n_k + \lambda_k. \tag{2.8}$$

The following example illustrates the conversion between different types of indices.

Example 2.4 Recalling the second part of Example 2.3, we may use different types of indices to label the elements.

1. Consider an element which is x_{11} in single-index form. Converting it into the order of $Id(\lambda_1, \lambda_2, \lambda_3; 2, 3, 4)$ by using (2.7), we have

$$\begin{aligned} p_3 &= p - 1 = 10, \\ \lambda_3 &= p_3 \% n_3 + 1 = 10\%4 + 1 = 2 + 1 = 3, \\ p_2 &= \left[\frac{p_3}{n_3}\right] = \left[\frac{10}{2}\right] = 2, \\ \lambda_2 &= p_2 \% n_2 + 1 = 2\%3 + 1 = 2 + 1 = 3, \end{aligned}$$

$$p_1 = \left[\frac{p_2}{n_2}\right] = \left[\frac{2}{4}\right] = 0,$$
$$\lambda_1 = p_1 \% n_1 + 1 = 0\%2 + 1 = 1.$$

Hence $x_{11} = x_{133}$.

2. Consider the element x_{214} in the order of $Id(\lambda_1, \lambda_2, \lambda_3; 2, 3, 4)$. Using (2.8), we have

$$p = (\lambda_1 - 1)n_2 n_3 + (\lambda_2 - 1)n_3 + \lambda_3 = 1 \cdot 3 \cdot 4 + 0 + 4 = 16.$$

Hence $x_{214} = x_{16}$.

3. In the order of $Id(\lambda_2, \lambda_3, \lambda_1; 3, 4, 2)$, the data are arranged as

$$\begin{array}{cccc} x_{111} & x_{211} & x_{112} & x_{212} \\ x_{113} & x_{213} & x_{114} & x_{214} \\ \vdots & & & \\ x_{131} & x_{231} & x_{132} & x_{232} \\ x_{133} & x_{233} & x_{134} & x_{234}. \end{array}$$

For this index, if we want to use the formulas for conversion between natural multi-index and single index, we can construct an auxiliary natural multi-index $y_{\Lambda_1,\Lambda_2,\Lambda_3}$, where $\Lambda_1 = \lambda_2$, $\Lambda_2 = \lambda_3$, $\Lambda_3 = \lambda_1$ and $N_1 = n_2 = 3$, $N_2 = n_3 = 4$, $N_3 = n_1 = 2$. Then, $b_{i,j,k}$ is indexed by $(\Lambda_1, \Lambda_2, \Lambda_3)$ in the order of $Id(\Lambda_1, \Lambda_2, \Lambda_3; N_1, N_2, N_3)$. In this way, we can use (2.7) and (2.8) to convert the indices.

For instance, consider x_{124}. Let $x_{124} = y_{241}$. For y_{241}, using (2.7), we have

$$\begin{aligned} p &= (\Lambda_1 - 1)N_2 N_3 + (\Lambda_2 - 1)N_3 + \Lambda_3 \\ &= (2-1) \times 4 \times 2 + (4-1) \times 2 + 1 = 8 + 6 + 1 = 15. \end{aligned}$$

Hence

$$x_{124} = y_{241} = y_{15} = x_{15}.$$

Consider x_{17} again. Since $x_{17} = y_{17}$, using (2.6), we have

$$\begin{aligned} p_3 &= p - 1 = 16, & \Lambda_3 &= p_3 \% N_3 + 1 = 1, \\ p_2 &= [p_3/N_3] = 8, & \Lambda_2 &= p_2 \% N_2 + 1 = 1, \\ p_1 &= [p_2/N_2] = 2, & \Lambda_1 &= p_1 \% N_1 + 1 = 3. \end{aligned}$$

Hence $x_{17} = y_{17} = y_{311} = x_{131}$.

From the above argument one sees that a set of higher-dimensional data, labeled by a multi-index, can be converted into a set of 1-dimensional data, labeled by

Table 2.1 The prisoner's dilemma

$P_1\backslash P_2$	A_1	A_2
A_1	$-1, -1$	$-9, 0$
A_2	$0, -9$	$-6, -6$

single-index. A matrix, as a set of 2-dimensional data, can certainly be converted into a set of 1-dimensional data. Consider a matrix

$$A = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}.$$

The row-stacking form of A, denoted by $V_r(A)$, is a row-by-row arranged nm-vector, i.e.,

$$V_r(A) = (a_{11}, a_{12}, \dots, a_{1n}, \dots, a_{m1}, a_{m2}, \dots, a_{mn})^T. \tag{2.9}$$

The column-stacking form of A, denoted by $V_c(A)$, is the following nm-vector:

$$V_c(A) = (a_{11}, a_{21}, \dots, a_{m1}, \dots, a_{1n}, a_{2n}, \dots, a_{mn})^T. \tag{2.10}$$

From the definition it is clear that we have the following result.

Proposition 2.2

$$V_c(A) = V_r\bigl(A^T\bigr), \qquad V_r(A) = V_c\bigl(A^T\bigr). \tag{2.11}$$

Finally, we give an example for multidimensional data labeled by an ordered multi-index.

Example 2.5

1. Consider the so-called prisoner's dilemma [5]. Two suspects are arrested and charged with a crime and each prisoner has two possible strategies:

$$A_1: \quad \text{not confess (or be mum);} \qquad A_2: \quad \text{confess (or fink).}$$

The payoffs are described by a payoff bi-matrix, given in Table 2.1.

For instance, if prisoner P_1 chooses "mum" (A_1) and P_2 chooses "fink" (A_2), P_1 will be sentenced to jail for nine months and P_2 will be released. Now, if we denote by

$$r^i_{j,k}, \quad i = 1, 2, j = 1, 2, k = 1, 2,$$

the payoff of P_i as P_1 takes strategy j and P_2 takes strategy k, then $\{r^i_{j,k}\}$ is a set of 3-dimensional data. We may arrange it into a payoff matrix as

$$M_p = \begin{bmatrix} r^1_{11} & r^1_{12} & r^1_{21} & r^1_{22} \\ r^2_{11} & r^2_{12} & r^2_{21} & r^2_{22} \end{bmatrix} = \begin{bmatrix} -1 & -9 & 0 & -6 \\ -1 & 0 & -9 & -6 \end{bmatrix}. \tag{2.12}$$

2. Consider a game with n players. Player P_i has k_i strategies and the payoff of P_i as P_j takes strategy s_j, $j = 1, \dots, n$, is

$$r^i_{s_1,\dots,s_n}, \quad i = 1, \dots, n; s_j = 1, \dots, k_j, j = 1, \dots, n.$$

Then, $\{r^i_{s_1,\dots,s_n}\}$ is a set of $(n+1)$-dimensional data. Arranging it with i as the row index and its column by the ordered multi-index $Id(s_1, \dots, s_n; k_1, \dots, k_n)$, we have

$$M_{\mathrm{g}} = \begin{bmatrix} r^1_{11\cdots 1} & \cdots & r^1_{11\cdots k_n} & \cdots & r^1_{1k_2\cdots k_n} & \cdots & r^1_{k_1k_2\cdots k_n} \\ \vdots & & & & & & \\ r^n_{11\cdots 1} & \cdots & r^n_{11\cdots k_n} & \cdots & r^n_{1k_2\cdots k_n} & \cdots & r^n_{k_1k_2\cdots k_n} \end{bmatrix}. \tag{2.13}$$

M_{g} is called the payoff matrix of game g.

2.2 Semi-tensor Product of Matrices

We consider the conventional matrix product first.

Example 2.6 Let U and V be m- and n-dimensional vector spaces, respectively. Assume $F \in L(U \times V, \mathbb{R})$, that is, F is a bilinear mapping from $U \times V$ to $\mathbb{R}$. Denote by $\{u_1, \dots, u_m\}$ and $\{v_1, \dots, v_n\}$ the bases of U and V, respectively. We call $S = (s_{ij})$ the structure matrix of F, where

$$s_{ij} = F(u_i, v_j), \quad i = 1, \dots, m, \ j = 1, \dots, n.$$

If we let $X = \sum_{i=1}^m x_i u_i \in U$, otherwise written as $X = (x_1, \dots, x_m)^{\mathrm{T}} \in U$, and $Y = \sum_{i=1}^n y_i v_i \in V$, otherwise written as $Y = (y_1, \dots, y_n)^{\mathrm{T}} \in V$, then

$$F(X, Y) = X^{\mathrm{T}} S Y. \tag{2.14}$$

Denoting the rows of S by $S^1, \dots, S^m$, we can alternatively calculate F in two steps.

Step 1: Calculate $x_1 S^1, x_2 S^2, \dots, x_m S^m$ and take their sum.
Step 2: Multiply $\sum_{i=1}^m x_i S^i$ by Y (which is a standard inner product).

It is easy to check that this algorithm produces the same result. Now, in the first step it seems that we have $(S^1 \cdots S^n) \times X$. This calculation motivates a new algorithm, which is defined as follows.

Definition 2.5 Let T be an np-dimensional row vector and X a p-dimensional column vector. Split T into p equal blocks, named $T^1, \dots, T^p$, which are $1 \times n$ matrices. Define a left semi-tensor product, denoted by $\ltimes$, as

$$T \ltimes X = \sum_{i=1}^p T^i x_i \in \mathbb{R}^n. \tag{2.15}$$

Using this new product, we reconsider Example 2.6 and propose another algorithm.

Example 2.7 (Example 2.6 continued) We rearrange the structure constants of F into a row as

$$T : V_{\mathrm{r}}(S) = (s_{11}, \ldots, s_{1n}, \ldots, s_{m1}, \ldots, s_{mn}),$$

called the structure matrix of F. This is a row vector of dimension mn, labeled by the ordered multi-index $Id(i, j; m, n)$. The following algorithm provides the same result as (2.14):

$$F(X, Y) = T \ltimes X \ltimes Y. \tag{2.16}$$

It is easy to check the correctness of (2.16), but what is its advantage? Note that (2.16) realized the product of 2-dimensional data (a matrix) with 1-dimensional data by using the product of two sets of 1-dimensional data. If, in this product, 2-dimensional data can be converted into 1-dimensional data, we would expect that the same thing can be done for higher-dimensional data. If this is true, then (2.16) is superior to (2.14) because it allows the product of higher-dimensional data to be taken. Let us see one more example.

Example 2.8 Let U, V, and W be m-, n-, and t-dimensional vector spaces, respectively, and let $F \in L(U \times V \times W, \mathbb{R})$. Assume $\{u_1, \ldots, u_m\}$, $\{v_1, \ldots, v_n\}$, and $\{w_1, \ldots, w_t\}$ are the bases of U, V, and W, respectively. We define the structure constants as

$$s_{ijk} = F(u_i, v_j, w_k), \quad i = 1, \ldots, m,\ j = 1, \ldots, n,\ k = 1, \ldots, t.$$

The structure matrix S of F can be constructed as follows. Its data are labeled by the ordered multi-index $Id(i, j, k; m, n, t)$ to form an mnt-dimensional row vector as

$$S = (s_{111}, \ldots, s_{11t}, \ldots, s_{1n1}, \ldots, s_{1nt}, \ldots, s_{mn1}, \ldots, s_{mnt}).$$

Then, for $X \in U$, $Y \in V$, $Z \in W$, it is easy to verify that

$$F(X, Y, Z) = S \ltimes X \ltimes Y \ltimes Z.$$

Observe that in a semi-tensor product, $\ltimes$ can automatically find the "pointer" of different hierarchies and then perform the required computation.

It is obvious that the structure and algorithm developed in Example 2.8 can be used for any multilinear mapping. Unlike the conventional matrix product, which can generally treat only one- or two-dimensional data, the semi-tensor product of matrices can be used to deal with any finite-dimensional data.

Next, we give a general definition of semi-tensor product.

Definition 2.6

(1) Let $X = (x_1, \dots, x_s)$ be a row vector, $Y = (y_1, \dots, y_t)^{\mathrm{T}}$ a column vector.

Case 1: If t is a factor of s, say, $s = t \times n$, then the n-dimensional row vector defined as

$$X \ltimes Y := \sum_{k=1}^{t} X^k y_k \in \mathbb{R}^n \tag{2.17}$$

is called the left semi-tensor inner product of X and Y, where

$$X = \left(X^1, \dots, X^t\right), \quad X^i \in \mathbb{R}^n, i = 1, \dots, t.$$

Case 2: If s is a factor of t, say, $t = s \times n$, then the n-dimensional column vector defined as

$$X \ltimes Y := \sum_{k=1}^{t} x_k Y^k \in \mathbb{R}^n \tag{2.18}$$

is called the left semi-tensor inner product of X and Y, where

$$Y = \left(\left(Y^1\right)^{\mathrm{T}}, \dots, \left(Y^t\right)^{\mathrm{T}}\right)^{\mathrm{T}}, \quad Y^i \in \mathbb{R}^n, i = 1, \dots, t.$$

(2) Let $M \in \mathscr{M}_{m\times n}$ and $N \in \mathscr{M}_{p\times q}$. If n is a factor of p or p is a factor of n, then $C = M \ltimes N$ is called the left semi-tensor product of M and N, where C consists of $m \times q$ blocks as $C = (C^{ij})$, and

$$C^{ij} = M^i \ltimes N_j, \quad i = 1, \dots, m, \ j = 1, \dots, q,$$

where $M^i = \mathrm{Row}_i(M)$ and $N_j = \mathrm{Col}_j(N)$.

Remark 2.1

1. In the first item of Definition 2.6, if $t = s$, the left semi-tensor inner product becomes the conventional inner product. Hence, in the second item of Definition 2.6, if $n = p$, the left semi-tensor product becomes the conventional matrix product. Therefore, the left semi-tensor product is a generalization of the conventional matrix product. Equivalently, the conventional matrix product is a special case of the left semi-tensor product.
2. Throughout this book, the default matrix product is the left semi-tensor product, so we simply call it the "semi-tensor product" (or just "product").
3. Let $A \in \mathscr{M}_{m\times n}$ and $B \in \mathscr{M}_{p\times q}$. For convenience, when $n = p$, A and B are said to satisfy the "equal dimension" condition, and when $n = tp$ or $p = tn$, A and B are said to satisfy the "multiple dimension" condition.
4. When $n = tp$, we write $A \succ_t B$; when $p = tn$, we write $A \prec_t B$.
5. So far, the semi-tensor product is a generalization of the matrix product from the equal dimension case to the multiple dimension case.

Example 2.9

1. Let $X = [2\ -1\ 1\ 2]$, $Y = [-2\ 1]^{\mathrm{T}}$. Then

$$X \ltimes Y = \begin{bmatrix} 2 & -1 \end{bmatrix} \times (-2) + \begin{bmatrix} 1 & 2 \end{bmatrix} \times 1 = \begin{bmatrix} -3 & 4 \end{bmatrix}.$$

2. Let

$$X = \begin{bmatrix} 2 & 1 & -1 & 3 \\ 0 & 1 & 2 & -1 \\ 2 & -1 & 1 & 1 \end{bmatrix}, \qquad Y = \begin{bmatrix} -1 & 2 \\ 3 & 2 \end{bmatrix}.$$

Then

$$X \ltimes Y = \begin{bmatrix} (21) \times (-1) + (-13) \times 3 & (21) \times 2 + (-13) \times 2 \\ (01) \times (-1) + (2-1) \times 3 & (01) \times 2 + (2-1) \times 2 \\ (2-1) \times (-1) + (11) \times 3 & (2-1) \times 2 + (11) \times 2 \end{bmatrix}$$

$$= \begin{bmatrix} -5 & 8 & 2 & 8 \\ 6 & -4 & 4 & 0 \\ 1 & 4 & 6 & 0 \end{bmatrix}.$$

Remark 2.2

1. The dimension of the semi-tensor product of two matrices can be determined by deleting the largest common factor of the dimensions of the two factor matrices. For instance,

$$A_{p\times qr} \ltimes B_{r\times s} \ltimes C_{qst\times l} = (A \ltimes B)_{p\times qs} \ltimes C_{qst\times l} = (A \ltimes B \ltimes C)_{pt\times l}.$$

In the first product, r is deleted, and in the second product, qs is deleted. This is a generalization of the conventional matrix product: for the conventional matrix product, $A_{p\times s}B_{s\times q} = (AB)_{p\times q}$, where s is deleted.
2. Unlike the conventional matrix product, for the semi-tensor product even $A \ltimes B$ and $B \ltimes C$ are well defined, but $A \ltimes B \ltimes C = (A \ltimes B) \ltimes C$ may not be well defined. For instance, $A \in \mathscr{M}_{3\times 4}$, $B \in \mathscr{M}_{2\times 3}$, $C \in \mathscr{M}_{9\times 1}$.

In the conventional matrix product the equal dimension condition has certain physical interpretation. For instance, inner product, linear mapping, or differential of compound multiple variable function, etc. Similarly, the multiple dimension condition has its physical interpretation, e.g., the product of different-dimensional data, tensor product, etc.

We give one more example.

Example 2.10 Denote by Δ_k the set of columns of the identity matrix I_k, i.e.,

$$\Delta_k = \mathrm{Col}\{I_k\} = \left\{\delta_k^i \,\middle|\, i = 1, 2, \dots, k\right\}.$$

Define

$$\mathscr{L} = \left\{B \in \mathscr{M}_{2^m\times 2^n} \,\middle|\, m \geq 1, n \geq 0, \mathrm{Col}(B) \subset \Delta_{2^m}\right\}. \tag{2.19}$$

The elements of $\mathscr{L}$ are called logical matrices. It is easy to verify that the semi-tensor product $\ltimes : \mathscr{L} \times \mathscr{L} \to \mathscr{L}$ is always well defined. So, when we are considering matrices in $\mathscr{L}$, we have full freedom to use the semi-tensor product. (The formal definition of a logical matrix is given in the next chapter.)

Comparing the conventional matrix product, the tensor product, and the semi-tensor product of matrices, it is easily seen that there are significant differences between them. For the conventional matrix product, the product is element-to-element, for the tensor product, it is a product of one element to a whole matrix, while for the semi-tensor product, it is one element times a block of the other matrix. This is one reason why we call this new product the "semi-tensor product".

The following example shows that in the conventional matrix product, an illegal term may appear after some legal computations. This introduces some confusion into the otherwise seemingly perfect matrix theory. However, if we extend the conventional matrix product to the semi-tensor product, it becomes consistent again. This may give some support to the necessity of introducing the semi-tensor product.

Example 2.11 Let $X, Y, Z, W \in \mathbb{R}^n$ be column vectors. Since $Y^{\mathrm{T}}Z$ is a scalar, we have

$$\left(XY^{\mathrm{T}}\right)\left(ZW^{\mathrm{T}}\right) = X\left(Y^{\mathrm{T}}Z\right)W^{\mathrm{T}} = \left(Y^{\mathrm{T}}Z\right)\left(XW^{\mathrm{T}}\right) \in \mathscr{M}_n. \tag{2.20}$$

Again using the associative law, we have

$$\left(Y^{\mathrm{T}}Z\right)\left(XW^{\mathrm{T}}\right) = Y^{\mathrm{T}}(ZX)W^{\mathrm{T}}. \tag{2.21}$$

A problem now arises: What is ZX? It seems that the conventional matrix product is flawed.

If we consider the conventional matrix product as a particular case of the semi-tensor product, then we have

$$\left(XY^{\mathrm{T}}\right)\left(ZW^{\mathrm{T}}\right) = Y^{\mathrm{T}} \ltimes (Z \ltimes X) \ltimes W^{\mathrm{T}}. \tag{2.22}$$

It is easy to prove that (2.22) holds. Hence, when the conventional matrix product is extended to the semi-tensor product, the previous inconsistency disappears.

The following two examples show how to use the semi-tensor product to perform multilinear computations.

Example 2.12

1. Let $(V, *)$ be an algebra (refer to Definition 2.2) and $\{e_1, e_2, \ldots, e_n\}$ a basis of V. For any two elements in this basis we calculate the product as

$$e_i * e_j = \sum_{k=1}^{n} c_{ij}^k e_k, \quad i, j, k = 1, 2, \ldots, n. \tag{2.23}$$

We then have the structure constants $\{c_{ij}^k\}$. We arrange the constants into a matrix as follows:

$$M = \begin{bmatrix} c_{11}^1 & c_{12}^1 & \cdots & c_{1n}^1 & \cdots & c_{nn}^1 \\ c_{11}^2 & c_{12}^2 & \cdots & c_{1n}^2 & \cdots & c_{nn}^2 \\ \vdots & & & & & \\ c_{11}^n & c_{12}^n & \cdots & c_{1n}^n & \cdots & c_{nn}^n \end{bmatrix}. \tag{2.24}$$

M is called the structure matrix of the algebra.

Let $X, Y \in V$ be given as

$$X = \sum_{i=1}^n a_i e_i, \qquad Y = \sum_{i=1}^n b_i e_i.$$

If we fix the basis, then X, Y can be expressed in vector form as

$$X = (a_1, a_2, \ldots, a_n)^{\mathrm{T}}, \qquad Y = (b_1, b_2, \ldots, b_n)^{\mathrm{T}}.$$

In vector form, the vector product of X and Y can be simply calculated as

$$X * Y = M \ltimes X \ltimes Y. \tag{2.25}$$

2. Consider the cross product on $\mathbb{R}^3$. Its structure constants were obtained in Example 2.1. We can arrange them into a matrix as

$$M_{\mathrm{c}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{2.26}$$

Now, if

$$X = \frac{1}{\sqrt{3}} \begin{bmatrix} 1 \\ -1 \\ 1 \end{bmatrix}, \qquad Y = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix},$$

then we have

$$X \times Y = M_{\mathrm{c}} X Y = \begin{bmatrix} 0.4082 \\ 0.8165 \\ 0.4082 \end{bmatrix}.$$

When a multifold cross product is considered, this form becomes very convenient. For instance,

$$X \times \underbrace{Y \times \cdots \times Y}_{100} = M_{\mathrm{c}}^{100} X Y^{100} = \begin{bmatrix} 0.5774 \\ -0.5774 \\ 0.5774 \end{bmatrix}.$$

Example 2.13 Let $\phi \in \mathscr{T}_t^s(V)$. That is, ϕ is a tensor on V with covariant order s and contra-variant order t. Suppose that its structure constants are $\{c_{j_1,j_2,\dots,j_t}^{i_1,i_2,\dots,i_s}\}$. Arrange it into a matrix by using the ordered multi-index $Id(i_1, i_2, \dots, i_s; n)$ for columns and the ordered multi-index $Id(j_1, j_2, \dots, j_t; n)$ for rows. The matrix turns out to be

$$M_\phi = \begin{bmatrix} c_{11\cdots 1}^{11\cdots 1} & \cdots & c_{11\cdots 1}^{11\cdots n} & \cdots & c_{11\cdots 1}^{nn\cdots n} \\ c_{11\cdots 2}^{11\cdots 1} & \cdots & c_{11\cdots 2}^{11\cdots n} & \cdots & c_{11\cdots 2}^{nn\cdots n} \\ \vdots & & & & \\ c_{nn\cdots n}^{11\cdots 1} & \cdots & c_{nn\cdots n}^{11\cdots n} & \cdots & c_{nn\cdots n}^{nn\cdots n} \end{bmatrix}. \tag{2.27}$$

It is the structure matrix of the tensor ϕ. Now, assume $\omega_i \in V^*$, $i = 1, 2, \dots, t$, and $X_j \in V$, $j = 1, 2, \dots, s$, where ω_i are expressed as rows, and X_j are expressed as columns. Then

$$\phi(\omega_1, \dots, \omega_t, X_1, \dots, X_s) = \omega_t \omega_{t-1} \cdots \omega_1 M_\phi X_1 X_2 \cdots X_s, \tag{2.28}$$

where the product symbol $\ltimes$ is omitted.

Next, we define the power of a matrix. The definition is natural and was used in the previous example.

Definition 2.7 Given a matrix $A \in \mathscr{M}_{p\times q}$ such that $p\%q = 0$ or $q\%p = 0$, we define $A^n, n > 0$, inductively as

$$\begin{cases} A^1 = A, \\ A^{k+1} = A^k \ltimes A, \quad k = 1, 2, \dots. \end{cases}$$

Remark 2.3 It is easy to verify that the above A^n is well defined. Moreover, if $p = sq$, where $s \in \mathbb{N}$, then the dimension of A^k is $s^k q \times q$; if $q = sp$, then the dimension of A^k is $p \times s^k p$.

Example 2.14

1. If X is a row or a column, then according to Definition 2.7, X^n is always well defined. Particularly, when X, Y are columns, we have

$$X \ltimes Y = X \otimes Y. \tag{2.29}$$

When X, Y are rows,

$$X \ltimes Y = Y \otimes X. \tag{2.30}$$

In both cases,

$$X^k = \underbrace{X \otimes \cdots \otimes X}_{k}. \tag{2.31}$$

2. Let $X \in \mathbb{R}^n$, $Y \in \mathbb{R}^q$ be column vectors and $A \in \mathscr{M}_{m\times n}$, $B \in \mathscr{M}_{p\times q}$. Then,

$$(AX) \ltimes (BY) = (A \otimes B)(X \ltimes Y). \tag{2.32}$$

   Particularly,

$$(AX)^k = (\underbrace{A \otimes \cdots \otimes A}_{k})X^k. \tag{2.33}$$

3. Let $X \in \mathbb{R}^m$, $Y \in \mathbb{R}^p$ be row vectors and A, B be matrices (as in 2. above). Then

$$(XA) \ltimes (YB) = (X \ltimes Y)(B \otimes A). \tag{2.34}$$

   Hence,

$$(XA)^k = X^k(\underbrace{A \otimes \cdots \otimes A}_{k}). \tag{2.35}$$

4. Consider the set of real kth order homogeneous polynomials of $x \in \mathbb{R}^n$ and denote it by B_n^k. Under conventional addition and real number multiplication, B_n^k is a vector space. It is obvious that x^k contains a basis (x^k itself is not a basis because it contains redundant elements). Hence, every $p(x) \in B_n^k$ can be expressed as $p(x) = Cx^k$, where the coefficients $C \in \mathbb{R}^{n^k}$ are not unique. Note that here $x = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$ is a column vector.

In the rest of this section we describe some basic properties of the semi-tensor product.

Theorem 2.1 *As long as $\ltimes$ is well defined, i.e., the factor matrices have proper dimensions, then $\ltimes$ satisfies the following laws:*

1. *Distributive law:*

$$\begin{cases} F \ltimes (aG \pm bH) = aF \ltimes G \pm bF \ltimes H, \\ (aF \pm bG) \ltimes H = aF \ltimes H \pm bG \ltimes H, \quad a, b \in \mathbb{R}. \end{cases} \tag{2.36}$$

2. *Associative law:*

$$(F \ltimes G) \ltimes H = F \ltimes (G \ltimes H). \tag{2.37}$$

(We refer to Appendix B for the proof.)

The block multiplication law also holds for the semi-tensor product.

Proposition 2.3 *Assume $A \succ_t B$ (or $A \prec_t B$). Split A and B into blockwise forms as*

$$A = \begin{bmatrix} A^{11} & \cdots & A^{1s} \\ \vdots & & \vdots \\ A^{r1} & \cdots & A^{rs} \end{bmatrix}, \qquad B = \begin{bmatrix} B^{11} & \cdots & B^{1t} \\ \vdots & & \vdots \\ B^{s1} & \cdots & B^{st} \end{bmatrix}.$$

If we assume $A^{ik} \succ_t B^{kj}, \forall i, j, k$ *(correspondingly,* $A^{ik} \prec_t B^{kj}, \forall i, j, k$ *), then*

$$A \ltimes B = \begin{bmatrix} C^{11} & \cdots & C^{1t} \\ \vdots & & \vdots \\ C^{r1} & \cdots & C^{rt} \end{bmatrix}, \tag{2.38}$$

where

$$C^{ij} = \sum_{k=1}^{s} A^{ik} \ltimes B^{kj}.$$

Remark 2.4 We have mentioned that the semi-tensor product of matrices is a generalization of the conventional matrix product. That is, if we assume $A \in \mathscr{M}_{m\times n}$, $B \in \mathscr{M}_{p\times q}$, and $n = p$, then

$$A \ltimes B = AB.$$

Hence, in the following discussion the symbol $\ltimes$ will be omitted, unless we want to emphasize it. Throughout this book, unless otherwise stated, the matrix product will be the semi-tensor product, and the conventional matrix product is its particular case.

As a simple application of the semi-tensor product, we recall an earlier example.

Example 2.15 Recall Example 2.5. To use a matrix expression, we introduce the following notation. Let δ_n^i be the ith column of the identity matrix I_n. Denote by P the variable of players, where $P = \delta_n^i$ means $P = P_i$, i.e., the player under consideration is P_i. Similarly, denote by x_i the strategy chosen by the ith player, where $x_i = \delta_{k_i}^j$ means that the jth strategy of player i is chosen.

1. Consider the prisoner's dilemma. The payoff function can then be expressed as

$$r_p(P, x_1, x_2) = P^{\mathrm{T}} \ltimes M_p \ltimes x_1 \ltimes x_2, \tag{2.39}$$

   where M_p is the payoff matrix, as defined in (2.12).
2. Consider the general case. The payoff function is then

$$r_{\mathrm{g}}(P, x_1, x_2, \ldots, x_m) = P^{\mathrm{T}} \ltimes M_{\mathrm{g}} \ltimes_{i=1}^{n} x_i, \tag{2.40}$$

   where M_{g} is defined in (2.13).

2.3 Swap Matrix

One of the major differences between the matrix product and the scalar product is that the scalar (number) product is commutative but the matrix product is not. That is, in general,

$$AB \neq BA.$$

Since the semi-tensor product is a generalization of the conventional matrix product, it would be absurd to expect it to be commutative. Fortunately, with some auxiliary tools, the semi-tensor product has some "commutative" properties, called pseudo-commutative properties. In the sequel, we will see that the pseudo-commutative properties play important roles, such as separating coefficients from the variables, which makes it possible for the calculation of polynomials of multiple variables to be treated in a similar way as the calculation of polynomials of a single variable. The swap matrix is the key tool for pseudo-commutativity of the semi-tensor product.

Definition 2.8 A swap matrix $W_{[m,n]}$ is an $mn \times mn$ matrix, defined as follows. Its rows and columns are labeled by double index (i, j), the columns are arranged by the ordered multi-index $Id(i, j; m, n)$, and the rows are arranged by the ordered multi-index $Id(j, i; n, m)$. The element at position $[(I, J), (i, j)]$ is then

$$w_{(IJ),(ij)} = \delta_{i,j}^{I,J} = \begin{cases} 1, & I = i \text{ and } J = j, \\ 0, & \text{otherwise.} \end{cases} \tag{2.41}$$

Example 2.16

1. Letting $m = 2$, $n = 3$, the swap matrix $W_{[m,n]}$ can be constructed as follows. Using double index (i, j) to label its columns and rows, the columns of W are labeled by $Id(i, j; 2, 3)$, that is, $(11, 12, 13, 21, 22, 23)$, and the rows of W are labeled by $Id(j, i; 3, 2)$, that is, $(11, 21, 12, 22, 13, 23)$. According to (2.41), we have

$$\begin{array}{c} \quad\quad\quad (11)\ (12)\ (13)\ (21)\ (22)\ (23) \\ W_{[2,3]} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{matrix} (11) \\ (21) \\ (12) \\ (22) \\ (13) \\ (23) \end{matrix}. \end{array}$$

2. Consider $W_{[3,2]}$. Its columns are labeled by $Id(i, j; 3, 2)$, and its rows are labeled by $Id(j, i; 2, 3)$. We then have

$$\begin{array}{c} \quad\quad\quad (11)\ (12)\ (21)\ (22)\ (31)\ (32) \\ W_{[3,2]} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{matrix} (11) \\ (21) \\ (31) \\ (12) \\ (22) \\ (32) \end{matrix}. \end{array}$$

The swap matrix is a special orthogonal matrix. A straightforward computation shows the following properties.

Proposition 2.4

1. *The inverse and the transpose of a swap matrix are also swap matrices. That is,*

$$W_{[m,n]}^{\mathrm{T}} = W_{[m,n]}^{-1} = W_{[n,m]}. \tag{2.42}$$

2. *When $m = n$, (2.42) becomes*

$$W_{[n,n]} = W_{[n,n]}^{\mathrm{T}} = W_{[n,n]}^{-1}. \tag{2.43}$$

3.

$$W_{[1,n]} = W_{[n,1]} = I_n. \tag{2.44}$$

Since the case of $m = n$ is particularly important, for compactness, we denote it as

$$W_{[n]} := W_{[n,n]}.$$

From (2.42) it is clear that $W_{[m,n]}$ is an orthogonal matrix. This is because, when used as a linear mapping from $\mathbb{R}^{mn}$ to $\mathbb{R}^{mn}$, it changes only the positions of the elements but not the values.

A swap matrix can be used to convert the matrix stacking forms, as described in the following result.

Proposition 2.5 *Let $A \in \mathscr{M}_{m\times n}$. Then*

$$\begin{cases} W_{[m,n]} V_{\mathrm{r}}(A) = V_{\mathrm{c}}(A), \\ W_{[n,m]} V_{\mathrm{c}}(A) = V_{\mathrm{r}}(A). \end{cases} \tag{2.45}$$

For double-index-labeled data $\{a_{ij}\}$, if it is arranged by $Id(i, j; m, n)$, then the swap matrix $W_{[m,n]}$ can convert its arrangement to the order of $Id(j, i; n, m)$ and vice versa. This is what the "swap" refers to. This property can also be extended to the multiple index-case. We give a rigorous statement for this.

Corollary 2.1 *Let the data $\{a_{ij} \mid 1 \le i \le m, 1 \le j \le n\}$ be arranged by the ordered multi-index $Id(i, j; m, n)$ as a column X. Then*

$$Y = W_{[m,n]} X$$

is the same data $\{a_{ij}\}$ arranged in the order of $Id(j, i; n, m)$.

Example 2.17

1. Let $X = (x_{11}, x_{12}, x_{13}, x_{21}, x_{22}, x_{23})$. That is, $\{x_{ij}\}$ is arranged by the ordered multi-index $Id(i, j; 2, 3)$. A straightforward computation shows

$$Y = W_{[23]}X = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{21} \\ x_{22} \\ x_{23} \end{bmatrix} = \begin{bmatrix} x_{11} \\ x_{21} \\ x_{12} \\ x_{22} \\ x_{13} \\ x_{23} \end{bmatrix}.$$

   That is, Y is the rearrangement of the elements x_{ij} in the order of $Id(j, i; 3, 2)$.
2. Let $X = (x_1, x_2, \dots, x_m)^{\mathrm{T}} \in \mathbb{R}^m$, $Y = (y_1, y_2, \dots, y_n)^{\mathrm{T}} \in \mathbb{R}^n$. We then have

$$\begin{aligned} X \otimes Y &= (x_1y_1, x_1y_2, \dots, x_1y_n, \dots, x_my_1, x_my_2, \dots, x_my_n)^{\mathrm{T}}, \\ Y \otimes X &= (y_1x_1, y_1x_2, \dots, y_1x_m, \dots, y_nx_1, y_nx_2, \dots, y_nx_m)^{\mathrm{T}} \\ &= (x_1y_1, x_2y_1, \dots, x_my_1, \dots, x_1y_n, x_2y_n, \dots, x_my_n)^{\mathrm{T}}. \end{aligned}$$

   They both consist of $\{x_iy_j\}$. However, in $X \otimes Y$ the elements are arranged in the order of $Id(i, j; m, n)$, while in $Y \otimes X$ the elements are arranged in the order of $Id(j, i; n, m)$. According to Corollary 2.1 we have

$$Y \otimes X = W_{[m,n]}(X \otimes Y). \tag{2.46}$$

   It is easy to check that $XY = X \otimes Y$, so we have

$$YX = W_{[m,n]}XY. \tag{2.47}$$

The following proposition comes from the definition.

Proposition 2.6

1. *Let $X = (x_{ij})$ be a set of data arranged as a column vector by the ordered multi-index $Id(i, j; m, n)$. Then $W_{[m,n]}X$ is a column with the same data, arranged by the ordered multi-index $Id(j, i; n, m)$.*
2. *Let $\omega = (\omega_{ij})$ be a set of data arranged by the ordered multi-index $Id(i, j; m, n)$. Then $\omega W_{[n,m]}$ is a row with the same set of data, arranged by the ordered multi-index $Id(j, i; n, m)$.*

A swap matrix can be used for multiple-index-labeled data and can swap two special indices. This allows a very useful generalization of the previous proposition, which we now state as a theorem.

Theorem 2.2

1. *Let* $X = (x_{i_1,\dots,i_k})$ *be a column vector with its elements arranged by the ordered multi-index* $Id(i_1,\dots,i_k; n_1,\dots,n_k)$. *Then*

$$[I_{n_1+\cdots+n_{t-1}} \otimes W_{[n_t,n_{t+1}]} \otimes I_{n_{t+2}+\cdots+n_k}]X$$

is a column vector consisting of the same elements, arranged by the ordered multi-index $Id(i_1,\dots,i_{t+1},i_t,\dots,t_k; n_1,\dots,n_{t+1},n_t,\dots,n_k)$.

2. *Let* $\omega = (\omega_{i_1,\dots,i_k})$ *be a row vector with its elements arranged by the ordered multi-index* $Id(i_1,\dots,i_k; n_1,\dots,n_k)$. *Then*

$$\omega[I_{n_1+\cdots+n_{t-1}} \otimes W_{[n_{t+1},n_t]} \otimes I_{n_{t+2}+\cdots+n_k}]$$

is a row vector consisting of the same elements, arranged by the ordered multi-index $Id(i_1,\dots,i_{t+1},i_t,\dots,t_k; n_1,\dots,n_{t+1},n_t,\dots,n_k)$.

$W_{[m,n]}$ can be constructed in an alternative way which is convenient in some applications. Denoting by δ_n^i the ith column of the identity matrix I_n, we have the following.

Proposition 2.7

$$W_{[m,n]} = \begin{bmatrix} \delta_n^1 \ltimes \delta_m^1 & \cdots & \delta_n^n \ltimes \delta_m^1 & \cdots & \delta_n^1 \ltimes \delta_m^m & \cdots & \delta_n^n \ltimes \delta_m^m \end{bmatrix}. \tag{2.48}$$

For convenience, we provide two more forms of swap matrix:

$$W_{[m,n]} = \begin{bmatrix} I_m \otimes {\delta_n^1}^{\mathrm{T}} \\ \vdots \\ I_m \otimes {\delta_n^n}^{\mathrm{T}} \end{bmatrix} \tag{2.49}$$

and, similarly,

$$W_{[m,n]} = \begin{bmatrix} I_n \otimes \delta_m^1, \dots, I_n \otimes \delta_m^m \end{bmatrix}. \tag{2.50}$$

The following factorization properties reflect the blockwise permutation property of the swap matrix.

Proposition 2.8 *The swap matrix has the following factorization properties*:

$$W_{[p,qr]} = (I_q \otimes W_{[p,r]})(W_{[p,q]} \otimes I_r) = (I_r \otimes W_{[p,q]})(W_{[p,r]} \otimes I_q), \tag{2.51}$$

$$W_{[pq,r]} = (W_{[p,r]} \otimes I_q)(I_p \otimes W_{[q,r]}) = (W_{[q,r]} \otimes I_p)(I_q \otimes W_{[p,r]}). \tag{2.52}$$

2.4 Properties of the Semi-tensor Product

In this section some fundamental properties of the semi-tensor product of matrices are introduced. Throughout, it is easily seen that when the conventional matrix product is extended to the semi-tensor product, almost all its properties continue to hold. This is a significant advantage of the semi-tensor product.

Proposition 2.9 *Assuming that A and B have proper dimensions such that $\ltimes$ is well defined, then*

$$(A \ltimes B)^{\mathrm{T}} = B^{\mathrm{T}} \ltimes A^{\mathrm{T}}. \tag{2.53}$$

The following property shows that the semi-tensor product can be expressed by the conventional matrix product plus the Kronecker product.

Proposition 2.10

1. *If $A \in \mathscr{M}_{m\times np}$, $B \in \mathscr{M}_{p\times q}$, then*

$$A \ltimes B = A(B \otimes I_n). \tag{2.54}$$

2. *If $A \in \mathscr{M}_{m\times n}$, $B \in \mathscr{M}_{np\times q}$, then*

$$A \ltimes B = (A \otimes I_p)B. \tag{2.55}$$

(We refer to Appendix B for the proof.)

Proposition 2.10 is a fundamental result. Many properties of the semi-tensor product can be obtained through it. We may consider equations (2.54) and (2.55) as providing an alternative definition of the semi-tensor product. In fact, the name "semi-tensor product" comes from this proposition. Recall that for $A \in \mathscr{M}_{m\times n}$ and $B \in \mathscr{M}_{p\times q}$, their tensor product satisfies

$$A \otimes B = (A \otimes I_p)(I_n \otimes B). \tag{2.56}$$

Intuitively, it seems that the semi-tensor product takes the "left half" of the product in the right-hand side of (2.56) to form the product.

The following property may be considered as a direct corollary of Proposition 2.10.

Proposition 2.11 *Let A and B be matrices with proper dimensions such that $A \ltimes B$ is well defined. Then*:

1. *$A \ltimes B$ and $B \ltimes A$ have the same characteristic functions.*
2. $\mathrm{tr}(A \ltimes B) = \mathrm{tr}(B \ltimes A)$.
3. *If A and B are invertible, then $A \ltimes B \sim B \ltimes A$, where "$\sim$" stands for matrix similarity.*
4. *If both A and B are upper triangular (resp., lower triangular, diagonal, orthogonal) matrices, then $A \ltimes B$ is also an upper triangular (resp., lower triangular, diagonal, orthogonal) matrix.*
5. *If both A and B are invertible, then $A \ltimes B$ is also invertible. Moreover,*

$$(A \ltimes B)^{-1} = B^{-1} \ltimes A^{-1}. \tag{2.57}$$

6. *If* $A \prec_t B$, *then*

$$\det(A \ltimes B) = \big[\det(A)\big]^t \det(B). \tag{2.58}$$

If $A \succ_t B$, *then*

$$\det(A \ltimes B) = \det(A)\big[\det(B)\big]^t. \tag{2.59}$$

The following proposition shows that the swap matrix can also perform the swap of blocks in a matrix.

Proposition 2.12

1. *Assume*

$$A = (A_{11}, \dots, A_{1n}, \dots, A_{m1}, \dots, A_{mn}),$$

where each block has the same dimension and the blocks are labeled by double index $\{i, j\}$ *and arranged by the ordered multi-index* $Id(i, j; m, n)$. *Then*

$$AW_{[n,m]} = (A_{11}, \dots, A_{m1}, \dots, A_{1n}, \dots, A_{mn})$$

consists of the same set of blocks, which are arranged by the ordered multi-index $Id(j, i; n, m)$.

2. *Let*

$$B = \big(B_{11}^{\mathrm{T}}, \dots, B_{1n}^{\mathrm{T}}, \dots, B_{m1}^{\mathrm{T}}, \dots, B_{mn}^{\mathrm{T}}\big)^{\mathrm{T}},$$

where each block has the same dimension and the blocks are labeled by double index $\{i, j\}$ *and arranged by the ordered multi-index* $Id(i, j; m, n)$. *Then*

$$W_{[m,n]}B = \big(B_{11}^{\mathrm{T}}, \dots, B_{m1}^{\mathrm{T}}, \dots, B_{1n}^{\mathrm{T}}, \dots, M_{mn}^{\mathrm{T}}\big)^{\mathrm{T}}$$

consists of the same set of blocks, which are arranged by the ordered multi-index $Id(j, i; n, m)$.

The product of a matrix with an identity matrix I has some special properties.

Proposition 2.13

1. *Let* $M \in \mathcal{M}_{m \times pn}$. *Then*

$$M \ltimes I_n = M. \tag{2.60}$$

2. *Let* $M \in \mathcal{M}_{m \times n}$. *Then*

$$M \ltimes I_{pn} = M \otimes I_p. \tag{2.61}$$

3. *Let* $M \in \mathcal{M}_{pm \times n}$. *Then*

$$I_p \ltimes M = M. \tag{2.62}$$

4. *Let* $M \in \mathscr{M}_{m\times n}$. *Then*

$$I_{pm} \ltimes M = M \otimes I_p. \tag{2.63}$$

In the following, some linear mappings of matrices are expressed in their stacking form via the semi-tensor product.

Proposition 2.14 *Let* $A \in \mathscr{M}_{m\times n}$, $X \in \mathscr{M}_{n\times q}$, $Y \in \mathscr{M}_{p\times m}$. *Then*

$$V_{\mathrm{r}}(AX) = A \ltimes V_{\mathrm{r}}(X), \tag{2.64}$$

$$V_{\mathrm{c}}(YA) = A^{\mathrm{T}} \ltimes V_{\mathrm{c}}(Y). \tag{2.65}$$

Note that (2.64) is similar to a linear mapping over a linear space (e.g., $\mathbb{R}^n$). In fact, as X is a vector, (2.64) becomes a standard linear mapping.

Using (2.64) and (2.65), the stacking expression of a matrix polynomial may also be obtained.

Corollary 2.2 *Let* X *be a square matrix and* $p(x)$ *be a polynomial, expressible as* $p(x) = q(x)x + p_0$. *Then*

$$V_{\mathrm{r}}\big(p(X)\big) = q(X)V_{\mathrm{r}}(X) + p_0 V_{\mathrm{r}}(I). \tag{2.66}$$

Using linear mappings on matrices, some other useful formulas may be obtained [4].

Proposition 2.15 *Let* $A \in \mathscr{M}_{m\times n}$ *and* $B \in \mathscr{M}_{p\times q}$. *Then*

$$(I_p \otimes A)W_{[n,p]} = W_{[m,p]}(A \otimes I_p), \tag{2.67}$$

$$W_{[m,p]}(A \otimes B)W_{[q,n]} = (B \otimes A). \tag{2.68}$$

In fact, (2.67) can be obtained from (2.68).

Proposition 2.16 *Let* $X \in \mathscr{M}_{m\times n}$ *and* $A \in \mathscr{M}_{n\times s}$. *Then*

$$XA = \big(I_m \otimes V_{\mathrm{r}}^{\mathrm{T}}(I_s)\big)W_{[s,m]}A^{\mathrm{T}}V_{\mathrm{c}}(X). \tag{2.69}$$

Roughly speaking, a swap matrix can swap a matrix with a vector. This is sometimes useful.

Proposition 2.17

1. *Let* Z *be a* t*-dimensional row vector and* $A \in \mathscr{M}_{m\times n}$. *Then*

$$ZW_{[m,t]}A = AZW_{[n,t]} = A \otimes Z. \tag{2.70}$$

2. *Let* Y *be a* t*-dimensional column vector and* $A \in \mathscr{M}_{m\times n}$. *Then*

$$AW_{[t,n]}Y = W_{[t,m]}YA = A \otimes Y. \tag{2.71}$$

The following lemma is useful for simplifying some expressions.

Lemma 2.1 *Let* $A \in \mathcal{M}_{m\times n}$. *Then*

$$W_{[m,q]} \ltimes A \ltimes W_{[q,n]} = I_q \otimes A. \tag{2.72}$$

The semi-tensor product has some pseudo-commutative properties. The following are some useful pseudo-commutative properties. Their usefulness will become apparent later.

Proposition 2.18 *Suppose we are given a matrix* $A \in \mathcal{M}_{m\times n}$.

1. *Let* $Z \in \mathbb{R}^t$ *be a column vector. Then*

$$AZ^{\mathrm{T}} = Z^{\mathrm{T}} W_{[m,t]} A W_{[t,n]} = Z^{\mathrm{T}}(I_t \otimes A). \tag{2.73}$$

2. *Let* $Z \in \mathbb{R}^t$ *be a column vector. Then*

$$ZA = W_{[m,t]} A W_{[t,n]} Z = (I_t \otimes A)Z. \tag{2.74}$$

3. *Let* $X \in \mathbb{R}^m$ *be a row vector. Then*

$$X^{\mathrm{T}} A = \left[V_{\mathrm{r}}(A)\right]^{\mathrm{T}} X. \tag{2.75}$$

4. *Let* $Y \in \mathbb{R}^n$ *be a row vector. Then*

$$AY = Y^{\mathrm{T}} V_{\mathrm{c}}(A). \tag{2.76}$$

5. *Let* $X \in \mathbb{R}^m$ *be a column vector and* $Y \in \mathbb{R}^n$ *a row vector. Then*

$$XY = Y W_{[m,n]} X. \tag{2.77}$$

Proposition 2.19 *Let* $A \in \mathcal{M}_{m\times n}$ *and* $B \in \mathcal{M}_{s\times t}$. *Then*

$$A \otimes B = W_{[s,m]} \ltimes B \ltimes W_{[m,t]} \ltimes A = (I_m \otimes B) \ltimes A. \tag{2.78}$$

Example 2.18 Assume

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}, \qquad B = \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \\ b_{31} & b_{32} \end{bmatrix},$$

where $m = n = 2$, $s = 3$ and $t = 2$. Then

$$W_{[3,2]} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$W_{[2,2]} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$W_{[3,2]} \ltimes B \ltimes W_{[2,2]} \ltimes A = \begin{bmatrix} b_{11} & b_{12} & 0 & 0 \\ b_{21} & b_{22} & 0 & 0 \\ b_{31} & b_{32} & 0 & 0 \\ 0 & 0 & b_{11} & b_{12} \\ 0 & 0 & b_{21} & b_{22} \\ 0 & 0 & b_{31} & b_{32} \end{bmatrix} \ltimes \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

$$= \begin{bmatrix} a_{11}b_{11} & a_{11}b_{12} & a_{12}b_{11} & a_{12}b_{12} \\ a_{11}b_{21} & a_{11}b_{22} & a_{12}b_{21} & a_{12}b_{22} \\ a_{11}b_{31} & a_{11}b_{32} & a_{12}b_{31} & a_{12}b_{32} \\ a_{21}b_{11} & a_{21}b_{12} & a_{22}b_{11} & a_{22}b_{12} \\ a_{21}b_{21} & a_{21}b_{22} & a_{22}b_{21} & a_{22}b_{22} \\ a_{21}b_{31} & a_{21}b_{32} & a_{22}b_{31} & a_{22}b_{32} \end{bmatrix} = A \otimes B.$$

As a corollary of the previous proposition, we have the following.

Corollary 2.3 *Let $C \in \mathcal{M}_{s\times t}$. Then, for any integer $m > 0$, we have*

$$W_{[s,m]} \ltimes C \ltimes W_{[m,t]} = I_m \otimes C. \tag{2.79}$$

Finally, we consider how to express a matrix in stacking form and vice versa, via the semi-tensor product.

Proposition 2.20 *Let $A \in \mathcal{M}_{m\times n}$. Then*

$$V_{\mathrm{r}}(A) = A \ltimes V_{\mathrm{r}}(I_n), \tag{2.80}$$

$$V_{\mathrm{c}}(A) = W_{[m,n]} \ltimes A \ltimes V_{\mathrm{c}}(I_n). \tag{2.81}$$

Conversely, we can retrieve A from its row- or column-stacking form.

Proposition 2.21 *Let $A \in \mathcal{M}_{m\times n}$. Then*

$$A = \big[I_m \otimes V_{\mathrm{r}}^{\mathrm{T}}(I_n)\big] \ltimes V_{\mathrm{r}}(A) = \big[I_m \otimes V_{\mathrm{r}}^{\mathrm{T}}(I_n)\big] \ltimes W_{[n,m]} \ltimes V_{\mathrm{c}}(A). \tag{2.82}$$

As an elementary application of semi-tensor product, we consider the following example.

Example 2.19 In mechanics, it is well known that the angular momentum of a rigid body about its mass center is

$$H = \int r \times (\omega \times r)\,\mathrm{d}m, \tag{2.83}$$

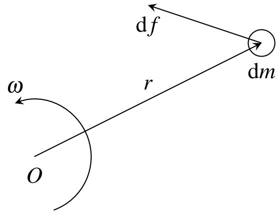

Fig. 2.1 Rotation

where $r=(x,y,z)$ is the position vector, starting from the mass center, and $\omega=(\omega_x,\omega_y,\omega_z)^{\mathrm{T}}$ is the angular speed. We want to prove the following equation for angular momentum (2.84), which often appears in the literature:

$$\begin{bmatrix} H_x \\ H_y \\ H_z \end{bmatrix} = \begin{bmatrix} I_x & -I_{xy} & -I_{zx} \\ -I_{xy} & I_y & -I_{yz} \\ I_{zx} & -I_{yz} & I_z \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \tag{2.84}$$

where

$$I_x=\int \left(y^2+z^2\right)\mathrm{d}m, \qquad I_y=\int \left(z^2+x^2\right)\mathrm{d}m, \qquad I_z=\int \left(x^2+y^2\right)\mathrm{d}m,$$
$$I_{xy}=\int xy\,\mathrm{d}m, \qquad I_{yz}=\int yz\,\mathrm{d}m, \qquad I_{zx}=\int zx\,\mathrm{d}m.$$

Let M be the moment of the force acting on the rigid body. We first prove that the dynamic equation of a rotating solid body is

$$\frac{\mathrm{d}H}{\mathrm{d}t}=M. \tag{2.85}$$

Consider a mass dm, with O as its rotating center, r as the position vector (from O to dm) and df the force acting on it (see Fig. 2.1). From Newton's second law,

$$\begin{aligned} \mathrm{d}f &= a\,\mathrm{d}m=\frac{\mathrm{d}v}{\mathrm{d}t}\,\mathrm{d}m \\ &= \frac{\mathrm{d}}{\mathrm{d}t}(\omega\times r)\,\mathrm{d}m. \end{aligned}$$

Now, consider the moment of force on it, which is

$$\mathrm{d}M=r\times \mathrm{d}f=r\times\frac{\mathrm{d}}{\mathrm{d}t}(\omega\times r)\,\mathrm{d}m.$$

Integrating this over the solid body, we have

$$M=\int r\times\frac{\mathrm{d}}{\mathrm{d}t}(\omega\times r)\,\mathrm{d}m. \tag{2.86}$$

We claim that

$$
\begin{aligned}
r \times \frac{\mathrm{d}}{\mathrm{d}t}(\omega \times r) &= \frac{\mathrm{d}}{\mathrm{d}t}\big[r \times (\omega \times r)\big], \\
RHS &= \frac{\mathrm{d}}{\mathrm{d}t}(r) \times (\omega \times r) + r \times \frac{\mathrm{d}}{\mathrm{d}t}(\omega \times r) \\
&= (\omega \times r) \times (\omega \times r) + r \times \frac{\mathrm{d}}{\mathrm{d}t}(\omega \times r) \\
&= 0 + r \times \frac{\mathrm{d}}{\mathrm{d}t}(\omega \times r) = LHS.
\end{aligned}
\tag{2.87}
$$

Applying this to (2.86), we have

$$
\begin{aligned}
M &= \int \frac{\mathrm{d}}{\mathrm{d}t}\big[r \times (\omega \times r)\big]\,\mathrm{d}m \\
&= \frac{\mathrm{d}}{\mathrm{d}t}\int r \times (\omega \times r)\,\mathrm{d}m \\
&= \frac{\mathrm{d}}{\mathrm{d}t}H.
\end{aligned}
$$

Next, we prove the angular momentum equation (2.84). Recall that the structure matrix of the cross product (2.26) is

$$
M_c = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},
$$

and for any two vectors $X, Y \in \mathbb{R}^3$, their cross product is

$$
X \times Y = M_c XY. \tag{2.88}
$$

Using this, we have

$$
\begin{aligned}
H &= \int r \times (\omega \times r)\,\mathrm{d}m \\
&= \int M_c r M_c \omega r\,\mathrm{d}m \\
&= \int M_c (I_3 \otimes M_c) r \omega r\,\mathrm{d}m \\
&= \int M_c (I_3 \otimes M_c) W_{[3,9]} r^2 \omega\,\mathrm{d}m \\
&= \int M_c (I_3 \otimes M_c) W_{[3,9]} r^2\,\mathrm{d}m\omega \\
&:= \int \Psi r^2\,\mathrm{d}m\omega,
\end{aligned}
$$

where

$$\begin{aligned}\Psi &= M_{\mathrm{c}}(I_3 \otimes M_{\mathrm{c}})W_{[3,9]}\\ &= \begin{bmatrix} 0&0&0&0&-1&0&0&0&-1&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\ 0&1&0&0&0&0&0&0&0&-1&0&0&0&0&0&0&0&-1&0&0&0&0&0&0&0&1&0\\ 0&0&1&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0&-1&0&0&0&-1&0&0&0&0 \end{bmatrix}.\end{aligned}$$

We then have

$$\Psi r^2 = \begin{bmatrix} y^2+z^2 & xy & -xz\\ -xy & x^2+z^2 & -yz\\ -xz & -yz & x^2+y^2 \end{bmatrix}.$$

Equation (2.84) follows immediately.

2.5 General Semi-tensor Product

In previous sections of this chapter the semi-tensor product considered was the left semi-tensor product of matrices. Throughout this book the default semi-tensor product is the left semi-tensor product, unless otherwise stated. In this section we will discuss some other kinds of semi-tensor products.

According to Proposition 2.10, an alternative definition of the left semi-tensor product is

$$A \ltimes B = \begin{cases} (A \otimes I_t)B, & A \prec_t B,\\ A(B \otimes I_t), & A \succ_t B. \end{cases} \tag{2.89}$$

This proceeds as follows. For a smaller-dimensional factor matrix, we match it on the right with an identity matrix of proper dimension such that the conventional matrix product is possible. The following definition then becomes natural.

Definition 2.9 Suppose we are given matrices A and B. Assuming $A \prec_t B$ or $A \succ_t B$, we define the right semi-tensor product of A and B as

$$A \rtimes B = \begin{cases} (I_t \otimes A)B, & A \prec_t B,\\ A(I_t \otimes B), & A \succ_t B. \end{cases} \tag{2.90}$$

Most properties of the left semi-tensor product hold for the right semi-tensor product. In Proposition 2.22 we assume the matrices have proper dimensions such the product $\rtimes$ is defined. In addition, for items 5–10 A and B are assumed to be two square matrices.

Proposition 2.22

1. *Associative law*:

$$(mA \rtimes B) \rtimes C = A \rtimes (B \rtimes C). \tag{2.91}$$

Distributive law:

$$(A+B)\ltimes C=A\ltimes C+B\ltimes C, \qquad C\ltimes(A+B)=C\ltimes A+C\ltimes B. \quad (2.92)$$

2. *Let X and Y be column vectors. Then*,

$$X\rtimes Y=Y\otimes X. \quad (2.93)$$

Let X and Y be row vectors. Then,

$$X\rtimes Y=X\otimes Y. \quad (2.94)$$

3.

$$(A\rtimes B)^{\mathrm{T}}=B^{\mathrm{T}}\rtimes A^{\mathrm{T}}. \quad (2.95)$$

4. *Let $M\in\mathscr{M}_{m\times pn}$. Then*,

$$M\rtimes I_n=M. \quad (2.96)$$

Let $M\in\mathscr{M}_{m\times n}$. Then,

$$M\rtimes I_{pn}=I_p\otimes M. \quad (2.97)$$

Let $M\in\mathscr{M}_{pm\times n}$. Then,

$$I_p\rtimes M=M. \quad (2.98)$$

Let $M\in\mathscr{M}_{m\times n}$. Then,

$$I_{pm}\rtimes M=I_p\otimes M. \quad (2.99)$$

5. *$A\rtimes B$ and $B\rtimes A$ have the same characteristic function.*
6.

$$\mathrm{tr}(A\rtimes B)=\mathrm{tr}(B\rtimes A). \quad (2.100)$$

7. *If A and B are orthogonal (upper triangular, lower triangular) matrices, then so is $A\rtimes B$.*
8. *If A and B are invertible, then $A\rtimes B\sim B\rtimes A$.*
9. *If A and B are invertible, then*

$$(A\rtimes B)^{-1}=B^{-1}\rtimes A^{-1}. \quad (2.101)$$

10. *If $A\prec_t B$, then*

$$\det(A\rtimes B)=\big[\det(A)\big]^t\det(B). \quad (2.102)$$

If $A\succ_t B$, then

$$\det(A\rtimes B)=\det(A)\big[\det(B)\big]^t. \quad (2.103)$$

A question which naturally arises is whether we can define the right semi-tensor product in a similar way as in Definition 2.6, i.e., in a "row-times-column" way. The answer is that we cannot. In fact, a basic difference between the right and the left semi-tensor products is that the right semi-tensor product does not satisfy the block product law. The row-times-column rule is ensured by the block product law. This difference makes the left semi-tensor product more useful. However, it is sometimes convenient to use the right semi-tensor product.

We now consider some relationships between left and right semi-tensor products.

Proposition 2.23 *Let X be a row vector of dimension np, Y a column vector of dimension p. Then,*

$$X \rtimes Y = XW_{[p,n]} \ltimes Y. \tag{2.104}$$

Conversely, we also have

$$X \ltimes Y = XW_{[n,p]} \rtimes Y. \tag{2.105}$$

If $\dim(X) = p$ *and* $\dim(Y) = pn$, *then*

$$X \rtimes Y = X \ltimes W_{[n,p]}Y. \tag{2.106}$$

Conversely, we also have

$$X \ltimes Y = X \rtimes W_{[p,n]}Y. \tag{2.107}$$

In the following, we introduce the left and right semi-tensor products of matrices of arbitrary dimensions. This will not be discussed beyond this section since we have not found any meaningful use for semi-tensor products of arbitrary dimensions.

Definition 2.10 Let $A \in \mathscr{M}_{m\times n}$, $B \in \mathscr{M}_{p\times q}$, and $\alpha = \operatorname{lcm}(n, p)$ be the least common multiple of n and p. The left semi-tensor product of A and B is defined as

$$A \ltimes B = (A \otimes I_{\frac{\alpha}{n}})(B \otimes I_{\frac{\alpha}{p}}). \tag{2.108}$$

The right semi-tensor product of A and B is defined as

$$A \rtimes B = (I_{\frac{\alpha}{n}} \otimes A)(I_{\frac{\alpha}{p}} \otimes B). \tag{2.109}$$

Note that if $n = p$, then both the left and right semi-tensor products of arbitrary matrices become the conventional matrix product. When the dimensions of the two factor matrices satisfy the multiple dimension condition, they become the multiple dimension semi-tensor products, as defined earlier.

Proposition 2.24 *The semi-tensor products of arbitrary matrices satisfy the following laws*:

1. *Distributive law*:

$$(A+B) \ltimes C = (A \ltimes C) + (B \ltimes C), \tag{2.110}$$

$$(A+B) \rtimes C = (A \rtimes C) + (B \rtimes C), \tag{2.111}$$

$$C \ltimes (A+B) = (C \ltimes A) + (C \ltimes B), \tag{2.112}$$

$$C \rtimes (A+B) = (C \rtimes A) + (C \rtimes B). \tag{2.113}$$

2. *Associative law*:

$$(A \ltimes B) \ltimes C = A \ltimes (B \ltimes C), \tag{2.114}$$

$$(A \rtimes B) \rtimes C = A \rtimes (B \rtimes C). \tag{2.115}$$

Almost all of the properties of the conventional matrix product hold for the left or right semi-tensor product of arbitrary matrices. For instance, we have the following.

Proposition 2.25

1.

$$\begin{cases} (A \ltimes B)^{\mathrm{T}} = B^{\mathrm{T}} \ltimes A^{\mathrm{T}}, \\ (A \rtimes B)^{\mathrm{T}} = B^{\mathrm{T}} \rtimes A^{\mathrm{T}}. \end{cases} \tag{2.116}$$

2. *If* $M \in \mathcal{M}_{m\times pn}$, *then*

$$\begin{cases} M \ltimes I_n = M, \\ M \rtimes I_n = M. \end{cases} \tag{2.117}$$

If $M \in \mathcal{M}_{pm\times n}$, *then*

$$\begin{cases} I_m \ltimes M = M, \\ I_m \rtimes M = M. \end{cases} \tag{2.118}$$

In the following, A *and* B *are square matrices*.

3. $A \ltimes B$ *and* $B \ltimes A$ ($A \rtimes B$ *and* $B \rtimes A$) *have the same characteristic function*.
4.

$$\begin{cases} \mathrm{tr}(A \ltimes B) = \mathrm{tr}(B \ltimes A), \\ \mathrm{tr}(A \rtimes B) = \mathrm{tr}(B \rtimes A). \end{cases} \tag{2.119}$$

5. *If both* A *and* B *are orthogonal* (*resp*., *upper triangular*, *lower triangular*, *diagonal*) *matrices*, *then* $A \ltimes B$ ($A \rtimes B$) *is orthogonal* (*resp*., *upper triangular*, *lower triangular*, *diagonal*).
6. *If both* A *and* B *are invertible*, *then* $A \ltimes B \sim B \ltimes A$ ($A \rtimes B \sim B \rtimes A$).
7. *If both* A *and* B *are invertible*, *then*

$$\begin{cases} (A \ltimes B)^{-1} = B^{-1} \ltimes A^{-1}, \\ (A \rtimes B)^{-1} = B^{-1} \rtimes A^{-1}. \end{cases} \tag{2.120}$$

8. *The determinant of the product satisfies*

$$\begin{cases} \det(A \ltimes B) = [\det(A)]^{\frac{\alpha}{n}} [\det(B)]^{\frac{\alpha}{p}}, \\ \det(A \rtimes B) = [\det(A)]^{\frac{\alpha}{n}} [\det(B)]^{\frac{\alpha}{p}}. \end{cases} \tag{2.121}$$

Corollary 2.4 *Let $A \in \mathscr{M}_{m\times n}$, $B \in \mathscr{M}_{p\times q}$. Then*

$$C = A \ltimes B = \left(C^{ij}\right), \quad i = 1, \ldots, m, j = 1, \ldots, q, \tag{2.122}$$

where

$$C^{ij} = A^i \ltimes B_j,$$

$A^i = \mathrm{Row}_i(A)$, *and* $B_j = \mathrm{Col}_j(B)$.

References

1. Bates, D., Watts, D.: Relative curvature measures of nonlinearity. J. R. Stat. Soc. Ser. B (Methodol.) **42**, 1–25 (1980)
2. Bates, D., Watts, D.: Parameter transformations for improved approximate confidence regions in nonlinear least squares. Ann. Stat. **9**, 1152–1167 (1981)
3. Cheng, D.: Sime-tensor product of matrices and its applications: a survey. In: Proc. 4th International Congress of Chinese Mathematicians, pp. 641–668. Higher Edu. Press, Int. Press, Hangzhou (2007)
4. Cheng, D., Qi, H.: Semi-tensor Product of Matrices—Theory and Applications. Science Press, Beijing (2007) (in Chinese)
5. Gibbons, R.: A Primer in Game Theory. Prentice Hall, New York (1992)
6. Lang, S.: Algebra, 3rd edn. Springer, New York (2002)
7. Mei, S., Liu, F., Xue, A.: Semi-tensor Product Approach to Transient Analysis of Power Systems. Tsinghua Univ. Press, Beijing (2010) (in Chinese)
8. Tsai, C.: Contributions to the design and analysis of nonlinear models. Ph.D. thesis, Univ. of Minnesota (1983)
9. Wang, X.: Parameter Estimate of Nonlinear Models—Theory and Applications. Wuhan Univ. Press, Wuhan (2002) (in Chinese)
10. Wei, B.: Second moments of LS estimate of nonlinear regressive model. J. Univ. Appl. Math. **1**, 279–285 (1986) (in Chinese)

Springer
http://www.springer.com/978-0-85729-096-0

Analysis and Control of Boolean Networks
A Semi-tensor Product Approach
Cheng, D.; Qi, H.; Li, Z.
2011, XVI, 472 p., Hardcover
ISBN: 978-0-85729-096-0