Chapter 3
Conflict Models in Graph Form

As depicted in Fig. 2.1 in Sect. 2.2.2, many models are available for describing strategic conflicts. For example, in the left branch of Fig. 2.1, metagame analysis employs option form (Howard 1971) for recording a conflict, while in the right branch, normal form is often written using a tabular or matrix format for the case of two decision makers (DMs). For the Graph Model for Conflict Resolution (GMCR) listed at the bottom of the left branch in Fig. 2.1, the movements in one step by a given DM are captured within a directed graph for that DM. As mentioned in Sect. 2.2.2, the models for the approaches given on the left in Fig. 2.1 only require relative preference information for each DM, while those in the right branch need cardinal preferences.

As explained in Sect. 1.2.2 and portrayed in Fig. 1.1, the key ingredients in any conflict model are the DMs, states or scenarios that could take place, and the preferences of each DM. The main purpose of this chapter is to define in detail these main modeling components with respect to GMCR. Because smaller conflicts are often conveniently recorded using what is called normal form, this type of abstract game model is described in Sect. 3.1.1. A very flexible format for writing down small, medium, and large conflict is option form which is defined in Sect. 3.1.2. Moreover, the exact linkages of the normal and option forms to the graph model are explained in this chapter. A simple conflict over sustainable development is employed to show how these three forms are used in practice and the connections among them. Additionally, a small conflict written in graph form is used in Sect. 3.2 to illustrate a situation which cannot be captured by either the normal or option form. Finally, the graph model is developed in a new direction, called matrix representation of the graph model, which is given in Sect. 3.3 and constitutes an equivalent way to represent the graph model. In fact, the matrix representation for GMCR is utilized throughout Chaps. 3–9 for addressing a range of situations (Xu et al. 2007, 2009a, b, c, 2010a, b, c, d, 2011, 2013, 2014, Bernath Walker et al. 2013, Hou et al. 2015).

H. Xu et al., *Conflict Resolution Using the Graph Model: Strategic Interactions in Competition and Cooperation*, Studies in Systems, Decision and Control 153,
https://doi.org/10.1007/978-3-319-77670-5_3

3.1 Normal Form and Option Form

3.1.1 Normal Form

In game theory, normal form is a way of describing a game using a list of strategies for each DM, together with preference information. Its formal definition is as follows.

Definition 3.1 (*Game in Normal Form*) A game G in normal form is usually written as a triplet $G = \langle N, \{T_i\}_{i \in N}, \{u_i\}_{i \in N} \rangle$, where

- $N = \{1, 2, \cdots, n\}$ is a nonempty set of DMs;
- for each DM $i \in N$, T_i is the nonempty strategy set of DM i;
- for each DM $i \in N$, $u_i : T_1 \times T_2 \times \cdots \times T_n \to \mathbb{R}$ is the utility of DM i.

In the above definition, let t_{ik} be a specific strategy for DM i, where $t_{ik} \in T_i$ and $m_i = |T_i|$ denote the number of strategies for DM i in T_i. Then $t = (t_{1a}, t_{2b}, \cdots, t_{ir}, \cdots, t_{nw})$ is called a strategy profile, where $t_{1a} \in T_1$, $t_{2b} \in T_2, \cdots, t_{ir} \in T_i, \cdots,$ and $t_{nw} \in T_n$. The symbol "$\times$" indicates the Cartesian product for which $T_1 \times T_2 \times \cdots \times T_n$ represents the set of all strategy profiles. Each element or strategy profile in this Cartesian product set is formed by selecting one element from each T_j and all possible combinations of these selections are used to create the total set of strategy profiles. When comparing the normal form with the graph form defined later, a strategy profile is also called a state so the state set $S = T = T_1 \times T_2 \times \cdots \times T_n$. The "$u_i$" denotes the von Newmann–Morgenstern (1953) utility function for DM i. For a given DM, a utility function maps each state to a real number for which a higher number means more preferred. In many disputes, a DM is interested in whether a state is more preferred to another state but not by how much. Therefore, one often employs $s >_i q$ to express that DM i prefers state s to state q.

To calculate the stability of a state for a given DM according to the different types of stability definitions presented in Chap. 4, one must define the set of movements in one step controlled separately by each DM in the conflict. Hence, one can expand the definition of a game in normal form by explicitly defining movement among states as is done in Sect. 3.2. When using $s \in S$ to represent a strategy as is done in the next definition, let $s = (s_{1a}, s_{2b}, \cdots, s_{ir}, \cdots, s_{nw})$ which indicates the strategy that each DM controls to form state s. Equivalently, this means that $s \in T_1 \times T_2 \times \cdots \times T_n$.

Definition 3.2 (*Unilateral Move in Normal Form*) For a game in normal form, the set of states to which DM $i \in N$ can unilaterally cause the game to move from state $s \in S$ is defined as:

$$R_i(s) = \{q \in S : q_{il} \neq s_{il} \text{ for some } 1 \leq l \leq m_i \text{ and}$$

$$q_{jk} = s_{jk} \text{ for any } j \in N \setminus \{i\} \text{ and } 1 \leq k \leq m_j\},$$

where $s_{il}, q_{il} \in T_i$, $s_{jk}, q_{jk} \in T_j$, and $\setminus$ refers to "set subtraction".

In words, this definition means that for a state q to be a unilateral move by DM i from state s (i.e. $q \in R_i(s)$), the strategy for DM i in state q is different from that in state s (i.e. $q_{il} \neq s_{il}$) and the strategies of the other DMs (i.e. $N \setminus \{i\}$) in state q remain the same as in state s ($q_{jk} = s_{jk}$ for any $j \in N \setminus \{i\}$). When ascertaining stability, as explained later in Chap. 4, one also must determine unilateral improvements by a DM as now defined.

Definition 3.3 (*Unilateral Improvement in Normal Form*) For a game in normal form, the set of unilateral improvements from state $s \in S$ for DM $i \in N$ is defined as:

$$R_i^+(s) = \{q \in R_i(s) : u_i(q) > u_i(s)\},$$

where u_i is DM i's utility.

A clear way to write down the normal form for a two-DM game is to use what is called 'matrix' form, as displayed in Table 3.1. As can be seen, DM 1, on the left, controls the two strategies $T_1 = \{t_{11}, t_{12}\}$ depicted as rows while DM 2 is in charge of the two strategies $T_2 = \{t_{21}, t_{22}\}$ given as columns. Each of the four cells in the matrix is a state s_k for which DM 1 and DM 2 have selected a strategy and contains the utility values of the state for DM 1 and DM 2. As can be seen at the bottom of Table 3.1, a state can be written as a situation in which each DM selects a strategy. Hence, state $s_3 = (t_{12}, t_{21})$ and this state appears as the bottom left cell in Table 3.1 for which the utility values for DMs 1 and 2 are $u_1(s_3)$ and $u_2(s_3)$, respectively.

As specified in Definition 3.2, when a given DM unilaterally causes the conflict to move from one state to another, the strategies of the other DMs remain the same. Referring to Table 3.1, notice that if DM 2 remains fixed at strategy t_{21} in the left column of the matrix, then DM 1 can cause the conflict to move from state s_1 to s_3 by changing his or her strategy from t_{11} to t_{12}. Similarly, DM 1 can make the game proceed from state s_3 to s_1 by changing his strategy from t_{12} in $s_3 = (t_{12}, t_{21})$ to the strategy t_{11} to form state $s_1 = (t_{11}, t_{21})$. Moreover, if state s_1 is more preferred by DM 1 to state s_3 (i.e. $u_1(s_1) > u_1(s_3)$), then the unilateral move from s_3 to s_1 is also

Table 3.1 2 × 2 game in normal form

		DM 2	
		t_{21}	t_{22}
DM 1	t_{11}	s_1: $u_1(s_1), u_2(s_1)$	s_2: $u_1(s_2), u_2(s_2)$
	t_{12}	s_3: $u_1(s_3), u_2(s_3)$	s_4: $u_1(s_4), u_2(s_4)$

$s_1 = (t_{11}, t_{21})$, $s_2 = (t_{11}, t_{22})$, $s_3 = (t_{12}, t_{21})$, $s_4 = (t_{12}, t_{22})$

Table 3.2 Sustainable development game in normal form

		DM 2	
		SD	**NSD**
DM 1	**P**	**10, 5** s_1	**6, 2** s_2
	NP	**8, 7** s_3	**1, 4** s_4

s_1 = (P, SD), s_2 = (P, NSD), s_3 = (NP, SD), s_4 = (NP, NSD)

a unilateral improvement according to Definition 3.3. Therefore, $s_1 \in R_1(s_3)$, $s_3 \in R_1(s_1)$, and $s_1 \in R_1^+(s_3)$. Finally, when examining unilateral moves by DM 2, one must fix the strategy on row, of DM 1. For instance, if DM 1 remains at strategy t_{11} on the first row in Table 3.1, then DM 2 can unilaterally cause the conflict to move from state s_1 to s_2 and back again. Therefore, $s_2 \in R_2(s_1)$ and $s_1 \in R_2(s_2)$. If DM 2 prefers state s_1 more than s_2 (i.e. $u_2(s_1) > u_2(s_2)$), then $s_1 \in R_2^+(s_2)$.

Example 3.1 (*Sustainable Development Conflict in Normal Form*) A specific illustration of the general 2×2 game displayed in Table 3.1 is the sustainable development game shown in Table 3.2. This environmental dispute was proposed by Hipel (2001) to model a basic conflict which could arise between an environmental agency (DM 1) and a developer (DM 2).

Therefore, the set of DMs is given by

$$N = \{DM\ 1,\ DM\ 2\}.$$

DM 1 can be either proactive (P) in encouraging responsible behavior by the developer with respect to environmental issues or not proactive (NP). As can be seen in Table 3.2, DM 1, on the left, controls its two strategies depicted as rows, where the strategy set for DM 1 is

$$T_1 = \{proactive\ (P),\ not\ proactive\ (NP)\} = \{P,\ NP\}.$$

DM 2, the developer of the project under consideration, can practice sustainable development (SD) or not adhere to sustainable development (NSD), which are displayed as columns in Table 3.2. Thus, DM 2 controls the strategy set

$$T_2 = \{sustainable\ development\ (SD),\ not\ sustainable\ development\ (NSD)\}$$
$$= \{SD,\ NSD\}.$$

When each DM selects a strategy, a state is created. Each of the four cells in Table 3.2 is a state for which DM 1 and DM 2 have selected a strategy. For instance, the cell labeled as state s_2 in Table 3.2 is the situation for which DM 1 selects strategy P and DM 2 chooses strategy NSD to produce state $s_2 = (P, NSD)$. Accordingly, the set of states in the sustainable development conflict is

$$T = T_1 \times T_2 = \{P, NP\} \times \{SD, NSD\} = \{(P, SD), (P, NSD), (NP, SD), (NP, NSD)\}.$$

$$\text{If } s_1 = (P, SD), s_2 = (P, NSD), s_3 = (NP, SD), \text{ and } s_4 = (NP, NSD), \text{ then}$$

$$T = \{s_1, s_2, s_3, s_4\}.$$

The two numbers written in each cell of the matrix in Table 3.2 represent the preference or utility of DM 1 and DM 2, respectively, where a high number means more preferred. Specifically, the utility values of DM 1 are

$$u_1(s_1) = 10, u_1(s_2) = 6, u_1(s_3) = 8, \text{ and } u_1(s_4) = 1;$$

while the utility values of DM 2 are

$$u_2(s_1) = 5, u_2(s_2) = 2, u_2(s_3) = 7, \text{ and } u_2(s_4) = 4.$$

The utility values for each of the two DMs can be used to order the states from most to least preferred such that:

$$s_1 >_1 s_3 >_1 s_2 >_1 s_4 \text{ for DM 1; and}$$

$$s_3 >_2 s_1 >_2 s_4 >_2 s_2 \text{ for DM 2.}$$

Note that the most preferred state for DM 1 (the environmental agency) is state s_1 for which DM 1 is proactive (P) and DM 2 is practicing sustainable development (SD). The least preferable state for DM 2 is state s_2 for which DM 2 has a preference value of 2. As explained for the example in Table 3.1, to determine the unilateral moves for DM 1, one first fixes DM 2 at a specified column. Hence, in Table 3.2, suppose that DM 2 has chosen strategy SD in the left column. Then DM 1 can cause the game to move from state s_1 to s_3 by changing his strategy selection from P to NP. Therefore, $R_1(s_1) = \{s_3\}$ and since DM 1 can change his strategy from NP to P in the same column, $R_1(s_3) = \{s_1\}$. Likewise, when DM 2 selects strategy NSD, DM 1 controls the movement in the second column and, therefore, $R_1(s_2) = \{s_4\}$ and $R_1(s_4) = \{s_2\}$.

As was also explained above for Table 3.1, to ascertain the states to which DM 2 can unilaterally cause to the game to move, DM 1's strategy must remain the same as either the first or the second row in Table 3.2. Accordingly, DM 2's unilateral moves are $R_2(s_1) = \{s_2\}$ and $R_2(s_2) = \{s_1\}$ when DM 1 remains at strategy P on the first row, and $R_2(s_3) = \{s_4\}$ and $R_2(s_4) = \{s_3\}$ when DM 1 is fixed at strategy NP in the second row.

Similarly, the unilateral improvements of DM 1 from the four states, respectively, are

$$R_1^+(s_1) = \emptyset, R_1^+(s_2) = \emptyset, R_1^+(s_3) = \{s_1\}, \text{ and } R_1^+(s_4) = \{s_2\};$$

while the unilateral improvements of DM 2 from the four states are

$$R_2^+(s_1) = \emptyset, R_2^+(s_2) = \{s_1\}, R_2^+(s_3) = \emptyset, \text{ and } R_2^+(s_4) = \{s_3\}.$$

3.1.2 Option Form

In a strategic conflict, a DM usually controls various courses of actions which are referred to as options. Let n be the number of DMs and O_i denote the option set of DM i, where o_{ij} is DM i's jth option. Then, the set of all options in a conflict model is $O = \bigcup_{i \in N} O_i$ in which index i indicates which DM controls the options. It may also be expressed as $O = \{O_1, O_2, \cdots, O_i, \cdots, O_n\}$, where the number of options in O_i is h_i. When a given DM decides which of his or her options to select or not a specific strategy is formed.

Definition 3.4 (*Strategy in Option Form*) Let O_i denote the option set of DM i for $i \in N$ for which $o_{ij} \in O_i$. A strategy for DM i is a mapping $g : O_i \to \{0, 1\}$, such that for $j = 1, 2, \cdots, h_i$

$$g(o_{ij}) = \begin{cases} 1 & \text{if DM } i \text{ selects option } o_{ij}, \\ 0 & \text{otherwise,} \end{cases}$$

where o_{ij} is DM i's jth option.

One can assign $g(o_{ij})$ a value of 1 to indicate that DM i will select option o_{ij}. Similarly, $g(o_{ij}) = 0$ means that DM i will not choose this option. A state is formed when each DM has selected a specific strategy. In other words, for each option the DM controlling the option has decided whether or not he or she will choose it. The formal definition for a state is as follows.

Definition 3.5 (*State in Option Form*) Let $O = \bigcup_{i \in N} O_i$ be the set of all options in a conflict for $o_{ij} \in O_i$, $i = 1, 2, \cdots, n$. A state is a mapping $f : O \to \{0, 1\}$, such that for $i = 1, 2, \cdots, n$,

$$f(o_{ij}) = \begin{cases} 1 & \text{if DM } i \text{ selects option } o_{ij}, \\ 0 & \text{otherwise.} \end{cases} \tag{3.1}$$

Let h denote the total number of options available to the DMs. A state can be treated as an h-dimensional column vector consisting of having an element of 0 or 1.

Therefore f_s is used to express the h-dimensional column vector to denote state s. Hence, f_s may be written as $[(g^s(O_1))^T, \cdots, (g^s(O_i))^T, \cdots, (g^s(O_n))^T]^T$ in which $g^s(O_i)$ denotes DM i's strategy corresponding to state s for $i = 1, 2, \cdots, n$ and is an h_i-dimensional column vector whose elements are

$$g^s(o_{ij}) = \begin{cases} 1 & \text{if DM } i \text{ selects option } o_{ij}, \\ 0 & \text{otherwise,} \end{cases}$$

A concise way to represent the set of all possible states in a conflict is to use the concept of a power set written as $\{0, 1\}^O$, where O is the set of all options, each of which can be not chosen or selected as indicated by 0 or 1, respectively. Therefore, the set of all mathematically possible states in a conflict model is $\{0, 1\}^O$. In mathematics, given a set O, the power set of O, written as 2^O, is the set of all subsets of O. Then, the power set of O contains $2^{|O|} = 2^h$ elements. Every state s can also be equivalently expressed as a subset of O, for which the mapping f is defined by Eq. 3.1. Although $2^{|O|} = 2^h$ states are mathematically possible, only a part of them are feasible in practice due to various option constraints, as explained in Sect. 3.2.2. The symbol S is used to designate the set of feasible states.

The option form is especially useful for practical applications because it can readily handle conflicts having any finite numbers of DMs, each of whom controls a finite number of option or courses of action. Consequently, as is done throughout this book, often option form is employed for writing down a conflict as part of the GMCR methodology. Because the number of states is typically much larger than the number of options in a conflict, when option form is employed in practice, the user only has to supply the relatively short list of options, for which the states can be automatically generated using a computer program. The option form is formally defined as follows.

Definition 3.6 (*Game in Option Form*) A game G in option form is usually written as $G = \langle N, \{O_i\}_{i \in N}, S, \{\succ_i, \sim_i\}_{i \in N} \rangle$, where

- $N = \{1, 2, \cdots, n\}$ is a nonempty set of DMs;
- for each DM $i \in N$, O_i is the nonempty option set of DM i;
- $S = \{s_1, s_2, \cdots, s_m\}$ is a nonempty set of feasible states;
- for each DM $i \in N$, $\{\succ_i, \sim_i\}$ represents i's preference where $s_k \succ_i s_t$ means that DM i prefers state s_k to state s_t while $s_k \sim_i s_t$ indicates that DM i has equal preference for these two states or is indifferent between them.

Note that the precise mathematical properties of $\{\succ_i, \sim_i\}$ are given in Sect. 3.2.4.

Similarly, one can expand the definition of a game in option form by explicitly defining unilateral moves and unilateral improvements among states as is done in Sect. 3.1.1. Let $h_i = |O_i|$ denote the cardinality of DM i's option set O_i and f_s stand for the mapping from options in the set O to state s.

Definition 3.7 (*Unilateral Moves in Option Form*) For a game in option form, the set of unilateral moves of DM $i \in N$ from state $s \in S$ is defined as:

$$R_i(s) = \{q \in S : g^q(o_{il}) \neq g^s(o_{il}) \text{ for some } o_{il} \in O_i \text{ and}$$

$$g^q(o_{jk}) = g^s(o_{jk}) \text{ for any } j \in N \setminus \{i\}\},$$

where $1 \leq l \leq h_i$ and $1 \leq k \leq h_j$.

One can define unilateral improvements by a DM for the option form based on Definition 3.7.

Definition 3.8 (*Unilateral Improvement in Option Form*) For a game in option form, the set of unilateral improvements from state $s \in S$ for DM $i \in N$ is defined as:

$$R_i^+(s) = \{q \in R_i(s) \text{ and } q \succ_i s\}.$$

Example 3.2 (*Sustainable Development Conflict in Option Form*) The sustainable development game for Example 3.1 has two DMs:

$$N = \{DM\ 1, DM\ 2\}.$$

DM 1 has the single option

$$O_1 = \{o_{11}\} = \{proactive\ (P)\} \text{ while}$$

DM 2 controls the option

$$O_2 = \{o_{21}\} = \{sustainable\ development\ (SD)\}.$$

When each DM decides upon which of his options to select or not, a state is formed. Table 3.3 presents the option form of the sustainable development game introduced in Example 3.1. The left column in this table lists each of the two DMs followed by the option which it controls. The four columns of Ys or Ns given on the right in Table 3.3 constitute the set of the four feasible states in this dispute where

$$S = \{s_1, s_2, s_3, s_4\}.$$

Table 3.3 Sustainable development game in option form

DM 1: Environmental agency				
1. Proactive (P)	Y	Y	N	N
DM 2: Developer				
2. Sustainable development (SD)	Y	N	Y	N
States	s_1	s_2	s_3	s_4

Preferences $s_1 \succ_1 s_3 \succ_1 s_2 \succ_1 s_4$ for DM 1 and
$s_3 \succ_2 s_1 \succ_2 s_4 \succ_2 s_2$ for DM 2

Rather than use 0 or 1 to indicate whether or not an option is taken as is done in Definitions 3.4 and 3.5, a Y or N is utilized, respectively, since these letter symbols are easier to interpret. Specifically, a "Y" indicates that an option is selected by the DM controlling it while an "N" means that the option is not chosen. Therefore, DM 1's two strategies are being proactive (Y) or not (N) and DM 2's two strategies are practicing sustainable development (Y) or not (N). DM 1's two strategies can also be expressed by $g(o_{11}) = 1$ and $g(o_{11}) = 0$ according to Definition 3.4. Similarly, $g(o_{21}) = 1$ and $g(o_{21}) = 0$ represent DM 2's two strategies. A state is any combination of Y's and N's opposite all of these DMs' options. Hence, a state is formed after each DM selects a strategy, so there are four states in the sustainable development game, which are written

$$s_1 = \begin{pmatrix} Y \\ Y \end{pmatrix} \text{ or } s_1 = (Y\ Y)^T,$$

$$s_2 = \begin{pmatrix} Y \\ N \end{pmatrix} \text{ or } s_2 = (Y\ N)^T,$$

$$s_3 = \begin{pmatrix} N \\ Y \end{pmatrix} \text{ or } s_3 = (N\ Y)^T,$$

and

$$s_4 = \begin{pmatrix} N \\ N \end{pmatrix} \text{ or } s_4 = (N\ N)^T,$$

where $(Y\ N)^T$, for example, denotes DM 1 will select the proactive option and DM 2 will not choose sustainable development. At the bottom of Table 3.3, the states are ranked or ordered by preference for each of the two DMs from most preferred on the left to least preferred on the right. Because the states are ordered according to preference, this type of preference is referred to as ordinal preference (see Sect. 3.2.4). Moreover, since there are no equally preferred states for each of the DMs, the preferences are said to be strict ordinal.

In the literature, the symbols $\succ_i$ and $\sim_i$ are often not used when it is known that the states are ranked from most to least preferred for a given DM. Accordingly, when employing option form, the ordering of states for each DM is as shown below:

$$\begin{pmatrix} Y & N & Y & N \\ Y & Y & N & N \\ s_1 & s_3 & s_2 & s_4 \end{pmatrix} \text{ and } \begin{pmatrix} N & Y & N & Y \\ Y & Y & N & N \\ s_3 & s_1 & s_4 & s_2 \end{pmatrix}.$$

Ordering of states for DM 1 Ranking of states for DM 2

To determine the unilateral move or moves for DM 1, one first fixes DM 2's strategy. Hence, in Table 3.3, suppose that DM 2 has chosen strategy SD as indicated by the Y located opposite SD and directly above s_1. Then, DM 1 can unilaterally cause the game to move from state s_1 to s_3 by changing his strategy selection from Y to

N. Therefore, since $g^{s_3}(o_{11}) \neq g^{s_1}(o_{11})$ and $g^{s_3}(o_{21}) = g^{s_1}(o_{21})$, then $R_1(s_1) = \{s_3\}$ according to Definition 3.7. Because DM 1 can also change his strategy from N to Y, $R_1(s_3) = \{s_1\}$. Likewise, when DM 2 selects strategy N, DM 1 controls the movement from s_2 to s_4 or from s_4 to s_2. Accordingly, $R_1(s_2) = \{s_4\}$ and $R_1(s_4) = \{s_2\}$. Similarly, to ascertain the states to which DM 2 can unilaterally cause the game to move, DM 1's strategy must be fixed. Hence, DM 2's unilateral moves are $R_2(s_1) = \{s_2\}$ and $R_2(s_2) = \{s_1\}$ when DM 1 remains at strategy $g^s(o_{11}) = Y$ and $R_2(s_3) = \{s_4\}$ and $R_2(s_4) = \{s_3\}$ when DM 1 is fixed at strategy $g^s(o_{11}) = N$.

In a similar fashion, the unilateral improvements of DM 1 from the four states, respectively, are

$$R_1^+(s_1) = \emptyset \text{ and } R_1^+(s_3) = \{s_1\} \text{ because } s_1 \succ_1 s_3,$$

$$\text{as well as } R_1^+(s_2) = \emptyset \text{ and } R_1^+(s_4) = \{s_2\} \text{ since } s_2 \succ_1 s_4,$$

while the unilateral improvements of DM 2 from the four states are

$$R_2^+(s_1) = \emptyset \text{ and } R_2^+(s_2) = \{s_1\} \text{ since } s_1 \succ_2 s_2,$$

$$\text{as well as } R_2^+(s_3) = \emptyset \text{ and } R_2^+(s_4) = \{s_3\} \text{ because } s_3 \succ_2 s_4.$$

3.2 Graph Model

The normal form of Sect. 3.1.1 provides a means for easily determining the states in a game, especially for the situation in which there are only two DMs. The option form defined in Sect. 3.1.2 can be conveniently utilized for ascertaining the states for both simple and complex games. In particular, when recording a conflict, as is done in Table 3.3 for the sustainable development conflict, one simply writes in the left column of the table the name of each of the DMs followed by all of options that the DM controls. The set of feasible states can then be written by hand on the right side of the table using the Y-N notation, as is done in Table 3.3. For a conflict having a relatively large number of DMs and options, a computer program can be used to generate the mathematically possible set of states, from which any infeasible states can be easily removed, as explained later in Sect. 3.2.2. Whatever the case, one ends up with the set of feasible states over which each DM has her or his own relative preferences. The definition of the graph model starts with the assumption that the set of feasible states are already known, and, for example, may have been generated using option form.

Definition 3.9 (*Graph Model*) A *graph model* is a structure

$$G = \langle N, S, \{A_i, \succeq_i, i \in N\}\rangle,$$

where

- N is a nonempty, finite set, called the set of DMs.
- S is a nonempty, finite set, called the set of feasible states.
- For each DM i, $A_i \subseteq S \times S$ is DM i's set of oriented arcs, which contains the movements in one step controlled by DM i.
- Precise mathematical properties of the preference relation for DM i, $\succeq_i$, is presented in Sect. 3.2.4

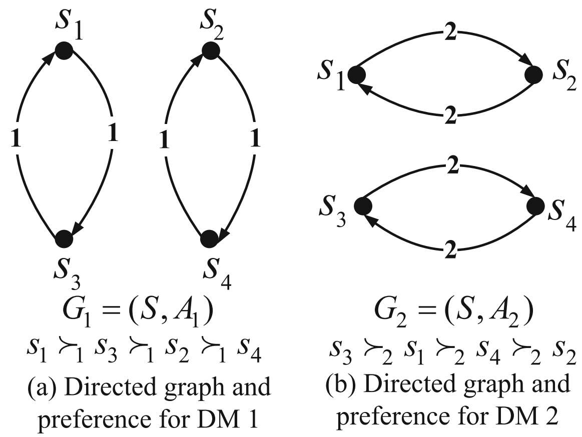
Note that $G_i = (S, A_i)$ is DM i's directed graph in which S denotes the vertex set and each oriented arc in $A_i \subseteq S \times S$ indicates that DM i can make a one-step unilateral move from the initial state of the arc to its terminal state. For simple models, sometimes it is informative to combine all of the DMs' directed graphs, $\{G_i : i \in N\}$, along with their preferences, to create what is called an integrated graph model.

Definition 3.10 The *integrated graph* of a graph model G is the structure $IG = \langle S, \{A_i, i \in N\}\rangle$.

In the above definition, an integrated graph IG, with vertex set S and arc set $A = \{A_i : i \in N\}$, contains all of the DM's individual graph $\{G_i : i \in N\}$. The arcs in A that are associated with DM i are considered to be labeled by i, or colored with color i. Thus, G may have multiple copies of an arc, but each copy has a different color. A two-DM conflict model is used to illustrate the components comprising a graph model.

Example 3.3 (*Sustainable Development Conflict in Graph Form*) The sustainable development game is first presented in Example 3.1. The graph model of this conflict is shown in Fig. 3.1, where the directed graph and relative preferences for DMs 1 and 2 are displayed on the left and right sides, respectively. For each DM, a given arc represents the unilateral movement, in one step, under that DM's control. Hence, for instance, DM 1 controls movement from state s_1 to state s_3, and back again, as

Fig. 3.1 Graph model for the sustainable development conflict



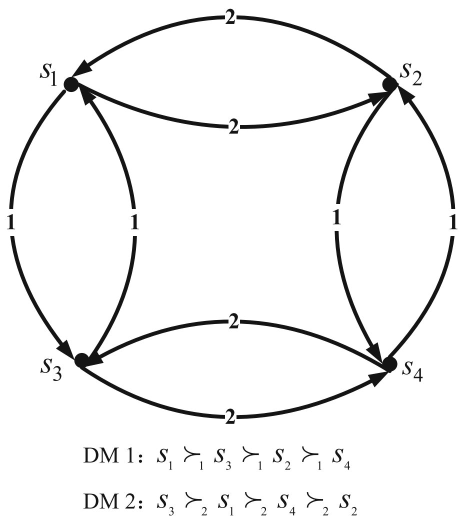


Fig. 3.2 Integrated graph model for the sustainable development conflict

indicated by the two arcs on the left in Fig. 3.1a, for which an arrow indicates the direction of movement between the two states.

Specifically, the graph model of the sustainable development conflict presents

- the DM set, $N = \{1, 2\}$;
- the state set, $S = \{s_1, s_2, s_3, s_4\}$;
- the directed graphs for the two DMs, $G_1 = (S, A_1)$ and $G_2 = (S, A_2)$ depicted in Fig. 3.1, where

$$A_1 = \{(s_1, s_3), (s_3, s_1), (s_2, s_4), (s_4, s_2)\} \text{ and } A_2 = \{(s_1, s_2), (s_2, s_1), (s_3, s_4), (s_4, s_3)\}.$$

- the preference information for the two DMs, which consists of

$$s_1 \succ_1 s_3 \succ_1 s_2 \succ_1 s_4 \text{ and } s_3 \succ_2 s_1 \succ_2 s_4 \succ_2 s_2;$$

The integrated graph IG, in combination with preference information, is called the integrated graph model. Figure 3.2 displays the integrated graph model for the sustainable development conflict.

In summary, a graph model contains the DMs, feasible states, the movements controlled by each DM which can be drawn as the set of separate directed graphs for the DMs or as a single integrated graph, and preference information. Although the normal form and option form can also represent the sustainable development game (see Sect. 3.1), they have a number of drawbacks. Particularly, movements among states in the normal and option formats are automatically restricted by their special structures. Figure 3.3 shows an example of a graph model that cannot be represented

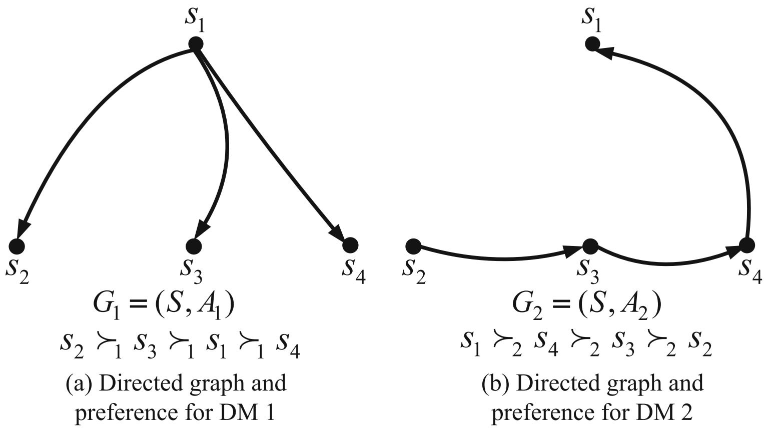


Fig. 3.3 A graph model

using the normal and option forms. Consider, for instance, Fig. 3.1a which is derived from the normal and option forms in Tables 3.2 and 3.3, respectively. Notice that in Fig. 3.1a, DM 1 cannot move from s_1 to s_2 and s_1 to s_4, as he or she can in Fig. 3.3a. Therefore, the states in Fig. 3.3 are derived in a way that permits this more general type of movement. In addition to permitting more flexible types of movement, the graph model possesses other advantages as discussed in Sects. 1.2 and 3.2.3 and elsewhere in the book.

3.2.1 Decision Makers

A strategic conflict is a situation in which two or more DMs with different objectives interact with one another. A DM may be an individual or a group, such as an industrial or governmental organization. For example, in a conflict in which family members are arguing over where to spend their next vacation, each DM is a person. In a trading conflict among car manufacturers which are trying to increase their shares of the automobile market, each DM represents a large company having many directors, shareholders, and employees. In previous research, a DM is also referred to as a player, actor, stakeholder, or participant. The term decision maker is used in this book, because it can stand for individuals or groups of people who can make decisions that affect a given conflict.

In the sustainable development conflict in Example 3.1, the environmental agency is a DM consisting of many people whose role is to protect the local environment from potential harm caused by the activities of a developer. Because the objective of the developer, which could be a large company, is to maximize profits, this DM's goal is in conflict with the aim of the environmental agency.

Any subset H of DMs in the set N is called a coalition. If $|H| > 0$, then the coalition H is nonempty. If $|H| > 1$, then the coalition H is nontrivial.

3.2.2 States

As mentioned before, states can be defined using normal form (Sect. 3.1.1) or option form (Sect. 3.1.2). Additionally, as shown by the types of movement in Fig. 3.3, which cannot be captured by the normal and option forms, states can be specified by other means. Nonetheless, option form is particularly useful for defining states in a rich range of real-world conflicts that can be readily investigated within the paradigm of the graph model. According to Definition 3.5, because each option can be either chosen or not, a conflict with h options has 2^h mathematically possible states. However, only a portion of them may be feasible in practice due to various option constraints. Additionally, each state can also be represented by a column indicating which options are selected (denoted by "Y") or not (indicated by "N"). As an example of a real-world dispute, consider the groundwater contamination conflict first mentioned in Sect. 1.2.2. In this dispute, Uniroyal Chemicals Ltd. (UR) polluted the aquifer underlying the town of Elmira located in Southern Ontario, Canada, from which the town previously obtained its water supplies. After the discovery of the pollutant, which is a carcinogen, the Ministry of the Environment (MoE) for the Province of Ontario, issued a Control Order in which it requested UR to treat its liquid discharges and cleanse the aquifer. The model in option form shown in Table 3.4 is for the negotiations that took place among the three DMs when UR appealed the Control Order which is a right it can exercise according to provincial law. Table 3.4, which is also given in Chap. 1 as Table 1.1, provides an explanation of the options controlled by the DMs. The feasible states for the negotiation are presented in Table 3.5.

Because each state can be either taken or not, a conflict having a total of five options as in Table 3.5 contains $2^5 = 32$ mathematically possible states. However, only the feasible states that could take place in reality are listed in Table 3.5 as columns. For convenience of explanation, each column or state is assigned a state number.

Table 3.4 Options for the Elmira model

MoE (Ministry of the Environment)
1. **Modify** the Control Order to make it more acceptable to UR
UR (Uniroyal Chemicals Ltd.)
2. **Delay** the appeal process
3. **Accept** the current Control Order
4. **Abandon** its Elmira operation
LG (Local Government)
5. **Insist** that the original Control Order be applied

Table 3.5 Feasible states for the Elmira model

MoE									
1. Modify	N	Y	N	Y	N	Y	N	Y	–
UR									
2. Delay	Y	Y	N	N	Y	Y	N	N	N
3. Accept	N	N	Y	Y	N	N	Y	Y	N
4. Abandon	N	N	N	N	N	N	N	N	Y
LG									
5. Insist	N	N	N	N	Y	Y	Y	Y	–
State number	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9

For instance, state s_5 is the scenario in which MoE is not modifying the Control Order, UR is delaying the negotiations but is not accepting the current Control Order and is not abandoning its factory in Elmira, and LG is insisting that the original Control Order be accepted by UR. As can be appreciated, UR's three options are mutually exclusive and, hence, UR cannot select two or more options at the same time. Therefore, any state in which UR chooses more than one of its three options is removed from the conflict since it is infeasible. Moreover, because UR is expected to do something, UR will choose at least one of its options. Finally, if UR abandons its plant, it does not matter what other options are selected by the other two DMs. Therefore, the set of resulting states are essentially the same and are represented as the single state s_9 in which a dash "–" indicates either Y or N.

3.2.3 State Transitions

One advantage of the graph model is its innate capability to systematically keep track of state transitions. State transition is the process by which a conflict model moves from one state to another. If a DM can cause a state transition on his or her own, then this transition is called a unilateral move (UM) for that DM. Let $R_i(s)$ denote DM i's reachable list from state s by UMs. This set contains all states to which DM i can move from state s in one step, and, hence, $R_i(s) = \{q \in S : (s, q) \in A_i\}$, where S is the set of feasible states and A_i is the set of arcs connecting two states which are controlled by DM i. For instance, in the sustainable development conflict shown in Fig. 3.2, Environment agency (DM 1) can move to state s_3 and Developer (DM 2) can reach state s_2 by one step from state s_1. Therefore, $R_1(s_1) = s_3$ and $R_2(s_1) = s_2$. Allowable state transitions constitute an important modeling component, as they determine the arc structure of a graph model and reflect the dynamic aspects of conflict in terms of potential moves and countermoves DMs can interactively take as they attempt to reach their goals.

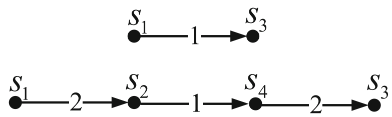


Fig. 3.4 Movements from state s_1 to state s_3 for the sustainable development conflict

In models based on option form, it is assumed that a DM has a UM from one state to another if and only if the two states differ in one or more options selected by that DM. In a graph model, moves controlled by each DM can be intuitively understood and seen as moves within each DM's directed graph or in combination within an integrated graph. The evolution of a conflict can be viewed as starting from a status quo (initial state) and then passing from one state to another, according to moves and countermoves controlled by individual DMs, until it eventually stops at some state such as an equilibrium or a compromise resolution. This interesting research topic about status quo analysis is discussed in Chap. 9. For example, all possible moves from state s_1 to state s_3 for the sustainable development conflict obtained from the integrated graph in Fig. 3.2 are depicted in Fig. 3.4. Note that an important restriction of a graph model is that no DM can move twice in succession along any path.

In the alternative normal form shown in Table 3.2, DM 1 may change the current position of the sustainable development conflict by changing the row but not the column, and DM 2 may change the column but not the row. For example, DM 1 can move from (P, SD) to (NP, SD), but not to (P, NSD) or (NP, NSD).

In option form, a DM can unilaterally cause the conflict to move from one state to another by changing his option choices when the other DM does not alter his option selections. For example, DM 1 can move unilaterally from $(\text{Y Y})^T$ to $(\text{N Y})^T$, but not to $(\text{Y N})^T$ or $(\text{N N})^T$.

Figure 3.5 shows the integrated graph for the Elmira model. By examining this figure or using an appropriate algorithm from Chap. 9, one can see that the following six possible paths connect state s_1 to s_8 where the letters on an arc indicate the DM controlling the movement between the two associated states for that arc.

$$s_1 \longrightarrow s_5 \longrightarrow s_6 \longrightarrow s_8,$$

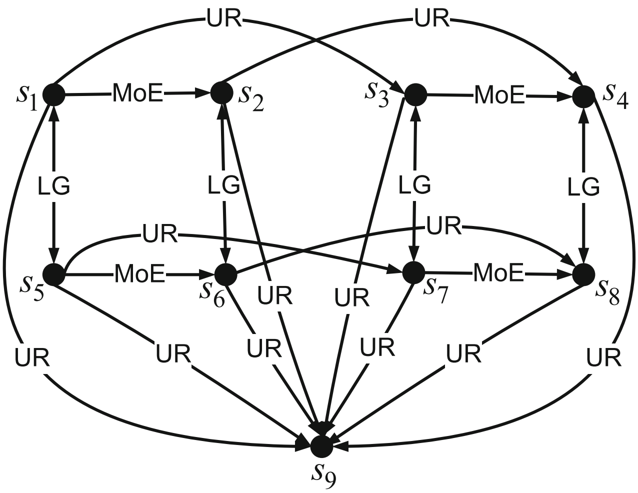
$$s_1 \longrightarrow s_5 \longrightarrow s_7 \longrightarrow s_8,$$

$$s_1 \longrightarrow s_3 \longrightarrow s_4 \longrightarrow s_8,$$

$$s_1 \longrightarrow s_2 \longrightarrow s_4 \longrightarrow s_8,$$

$$s_1 \longrightarrow s_3 \longrightarrow s_7 \longrightarrow s_8,$$

$$s_1 \longrightarrow s_2 \longrightarrow s_6 \longrightarrow s_8.$$



MoE: $s_7 \succ_1 s_3 \succ_1 s_4 \succ_1 s_8 \succ_1 s_5 \succ_1 s_1 \succ_1 s_2 \succ_1 s_6 \succ_1 s_9$

UR: $s_1 \succ_2 s_4 \succ_2 s_8 \succ_2 s_5 \succ_2 s_9 \succ_2 s_3 \succ_2 s_7 \succ_2 s_2 \succ_2 s_6$

LG: $s_7 \succ_3 s_3 \succ_3 s_5 \succ_3 s_1 \succ_3 s_8 \succ_3 s_6 \succ_3 s_4 \succ_3 s_2 \succ_3 s_9$

Fig. 3.5 Integrated graph for the Elmira model

If the name of each DM in an integrated graph is replaced by assigning a distinct color to any arc controlled by that DM, this produces what is called a colored graph (Xu et al. 2009b, 2013) (see Sect. 3.3.1).

3.2.4 Preferences

Obviously, preference information plays an important role in decision analysis. Each DM has preferences among the possible states that can arise. One way to express preferences is to use real numbers. For example, one object may have a monetary value of \$5 and another \$10. However, \$5 and \$10 may be worth much more to a poor person than a rich one. Therefore, the concept of utility theory was proposed to reflect the worth or utility of an object. More specifically, cardinal utility refers to a measurement scale for utility, often expressed as utils, that permits one to quantitatively compare the utility of objects. For the case of conflict resolution, utility values would reflect the preferences of a person or DM among the feasible states where a higher number means more preferred. The graph model requires only relative preference information for each DM, but can of course use cardinal information; moreover, it can handle both intransitive and transitive preferences. The formal definition for transitive preference is given below.

Definition 3.11 Let R denote any relation between two states. For any $k, s, q \in S$, if $k\ R\ s$ and $s\ R\ q$ imply $k\ R\ q$, then R is transitive.

The most basic type of preference is when two objects or states are compared in what is called a binary preference relationship. In the original graph model, simple preference (Fang et al. 1993) of DM i is coded by a pair of relations $\{\sim_i, \succ_i\}$ on S, where $s \succ_i q$ indicates that DM i prefers s to q and $s \sim_i q$ means that DM i is indifferent between s and q (or equally prefers s and q). Strict preference $\succ$ is transitive in many graph models, though in some cases it is intransitive. It is assumed that the preference relations of each DM $i \in N$ have the following properties:

(i) $\sim_i$ is reflexive and symmetric (i.e., $\forall s, q \in S$, $s \sim_i s$, and if $s \sim_i q$, then $q \sim_i s$);
(ii) $\succ_i$ is asymmetric (i.e., $s \succ_i q$ and $q \succ_i s$ cannot occur simultaneously);
(iii) $\{\sim_i, \succ_i\}$ is strongly complete.

Property (iii) implies that, for any $s, t \in S$, exactly one of the following statements is true: $s \succ_i t$, $t \succ_i s$, or $s \sim_i t$. The conventions that $s \succeq_i q$ is equivalent to either $s \succ_i q$ or $s \sim_i q$, and that $s \prec_i q$ is equivalent to $q \succ_i s$, are convenient.

If the definition for transitive preferences given in Definition 3.11 does not hold then the preferences are said to be intransitive. In Sect. 1.2.2 and Fig. 1.3, an example is provided for when a person, say DM i, compares three beverages according to preference. For the case of transitivity: Coffee $\succ_i$ Tea and Tea $\succ_i$ Coke implies Coffee $\succ_i$ Coke; For the case of intransitivity, the above relationship does not hold. Coffee $\succ_i$ Tea and Tea $\succ_i$ Coke but Coke $\succ_i$ Coffee. For the graph model, transitivity of preferences is not required, and all results hold whether preferences are transitive or intransitive.

The state set S can be divided into subsets based on preference relative to a fixed state $s \in S$. These subsets are essential components in stability analysis. The descriptions of these subsets for simple preference are presented as follows:

- $\Phi_i^+(s) = \{q : q \succ_i s\}$ denotes states preferred to state s by DM i;
- $\Phi_i^=(s) = \{q : q \sim_i s\}$ denotes states equally preferred to state s by DM i;
- $\Phi_i^-(s) = \{q : s \succ_i q\}$ denotes states less preferred than state s for DM i.

3.2.5 Directed Graph

A graph is a pair (V, E) of sets satisfying $E \subseteq V \times V$. A directed graph $G = (V, A, \psi)$, which is also called a digraph (Dieste 1997), is a set of vertices (nodes) V and a set of oriented edges (arcs) A with $\psi : A \to V \times V$. If $a \in A$ satisfies

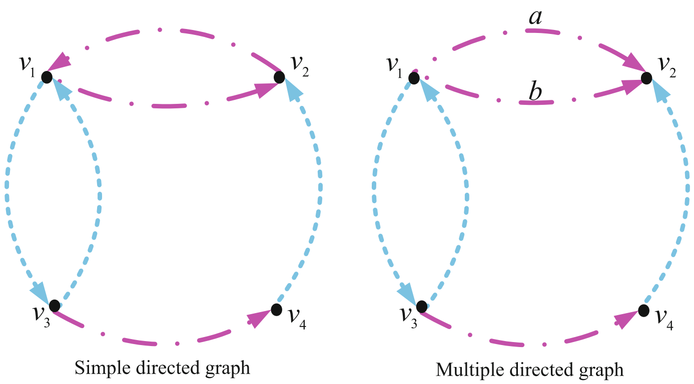


Fig. 3.6 Directed graphs

$\psi(a) = (u, v)$, then we say that a has initial vertex u and terminal vertex v. A multidigraph is a digraph containing multiple edges, i.e., it may contain $a, b \in A$ such that $a \neq b$ and $\psi(a) = \psi(b)$, in which case a and b are said to be multiple arcs. A digraph with no multiple edges is called a simple digraph (Dieste 1997). Figure 3.6 depicts a simple directed graph and a multidigraph. For the multidigraph, a and b are multiple edges, i.e., $\psi(a) = \psi(b) = (v_1, v_2)$. If there exists $a \in A$ such that $\psi(a) = (u, v)$, then u is said to be *adjacent* to v and (u, v) is said to be incident from u and incident to v. Hence, (u, v) is called in-incident to v and out-incident to u. When G is drawn, it is common to represent the direction of an edge with an arrowhead. One generally assumes loop-free graphs; i.e., for any $a \in A$, if $\psi(a) = (u, v)$, then $u \neq v$.

3.3 Matrix Representation of a Graph Model

It is well-known that matrices can efficiently describe adjacency of vertices, and incidence of arcs and vertices in a graph, thereby permitting tracking of paths between any two vertices (Godsil and Royle 2001). Matrices possess various algebraic properties, which can be exploited to develop improved algorithms for solving a variety of problems in a graph. As such, extensive research has been conducted to design effective algorithms and efficient search procedures by exploring relationships between matrices and paths (Gondran and Minoux 1979, Shiny and Pujari 1998, Hoffman and Schiebe 2001). Because a graph model consists of several interrelated graphs, it is natural to use well-known results of Algebraic Graph Theory to help to analyze it. The adjacency matrix can be applied to represent some directed graphs. However, if a graph model contains multiple arcs between the same two states controlled by different DMs, the adjacency matrix would be unable to track all aspects of conflict evolution from the status quo state. It is well known that the incidence matrix can represent multidigraphs if all edges are labeled (Godsil and Royle 2001).

It is common to combine all DMs' graphs, $\{G_i : i \in N\}$, into an *integrated* graph G with vertex set S and arc set $A = \cup\{A_i : i \in N\}$. The arcs in A that are associated with DM i are considered to be labeled by i, or colored with color i. Thus G may have multiple copies of an arc, but each copy is a different color. A unique edge-labeling rule for colored multidigraphs is proposed in the next subsection.

3.3.1 Definitions from Algebraic Graph Theory

Definition 3.12 For a digraph $G = (V, A, \psi)$, edge $a \in A$ and edge $b \in A$ **are consecutive (in the order** ab**)** iff $\psi(a) = (u, v)$ and $\psi(b) = (v, s)$, where $u, v, s \in V$.

Definition 3.13 For a digraph $G = (V, A, \psi)$, the **line digraph** $L(G) = (A, LA)$ of G is a simple digraph with vertex set A and edge set $LA=\{d = (a, b) \in A \times A : a$ and b are consecutive (in the order ab)$\}$.

An example is given in Fig. 3.7 with the directed graph and the line graph underneath it.

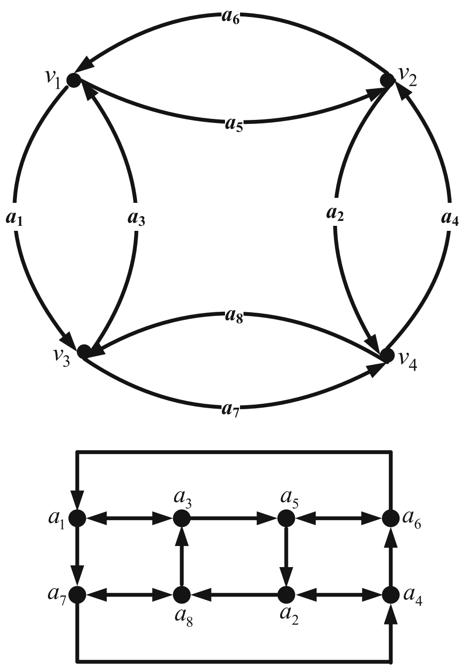


Fig. 3.7 A directed graph and its line graph

Definition 3.14 For a digraph $G = (V, A, \psi)$, a **path** from vertex $u \in V$ to vertex $s \in V$ is a sequence of vertices in G starting with u and ending with s, such that consecutive vertices are adjacent.

Note that in this book a path may contain the same vertex more than once (Buckley and Harary 1990). The length of a path is the number of edges therein.

Definition 3.15 For two $m \times m$ matrices M and Q, the **Hadamard product** for the two matrices is the $m \times m$ matrix $H = M \circ Q$ with (s, q) entry

$$H(s, q) = M(s, q) \cdot Q(s, q).$$

$$\text{If } M = \begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{pmatrix} \text{ and } Q = \begin{pmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \\ q_{41} & q_{42} & q_{43} & q_{44} \end{pmatrix},$$

$$\text{then } H = \begin{pmatrix} m_{11} \cdot q_{11} & m_{12} \cdot q_{12} & m_{13} \cdot q_{13} & m_{14} \cdot q_{14} \\ m_{21} \cdot q_{21} & m_{22} \cdot q_{22} & m_{23} \cdot q_{23} & m_{24} \cdot q_{24} \\ m_{31} \cdot q_{31} & m_{32} \cdot q_{32} & m_{33} \cdot q_{33} & m_{34} \cdot q_{34} \\ m_{41} \cdot q_{41} & m_{42} \cdot q_{42} & m_{43} \cdot q_{43} & m_{44} \cdot q_{44} \end{pmatrix}.$$

Let "$\vee$" denote the disjunction operator ("or") on two matrices. Assuming that H and G are two $m \times m$ matrices, the disjunction operation on matrices H and G is defined by:

Definition 3.16 For two $m \times m$ matrices H and G, **disjunction matrix** of H and G is the $m \times m$ matrix $M = H \vee G$ with (u, v) entry

$$M(u, v) = \begin{cases} 1 \text{ if } H(u, v) + G(u, v) \neq 0, \\ 0 \text{ otherwise.} \end{cases}$$

$$\text{If } H = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \text{ and } G = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix},$$

$$\text{then } M = H \vee G = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \end{pmatrix}.$$

Definition 3.17 The **sign function**, $sign(\cdot)$, maps an $m \times m$ matrix with (u, v) entry $M(u, v)$ to the $m \times m$ matrix

$$sign[M(u,v)] = \begin{cases} 1 & M(u,v) > 0, \\ 0 & M(u,v) = 0, \\ -1 & M(u,v) < 0. \end{cases}$$

If $M = \begin{pmatrix} 1.8 & 0 & -9.7 & 1 \\ 0 & -11.3 & 0 & 117.9 \\ -1.4 & 12.3 & 0 & 89.5 \\ 0 & -77.9 & 0 & 96.5 \end{pmatrix}$, then $sign(M) = \begin{pmatrix} 1 & 0 & -1 & 1 \\ 0 & -1 & 0 & 1 \\ -1 & 1 & 0 & 1 \\ 0 & -1 & 0 & 1 \end{pmatrix}$.

Important matrices associated with a digraph include the adjacency matrix and the incidence matrix (Godsil and Royle 2001). Let $m = |V|$ denote the number of vertices and $l = |A|$ be the number of edges of the directed graph G. Then,

Definition 3.18 For a multidigraph $G = (V, A, \psi)$, the **adjacency matrix** is the $m \times m$ matrix J with (u, v) entry

$$J(u,v) = \begin{cases} 1 & \text{if } (u,v) \in A, \\ 0 & \text{otherwise,} \end{cases}$$

where $u, v \in V$.

For the directed graph shown in Fig. 3.7, the adjacency matrix is expressed as

$$J = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}.$$

The adjacency matrix is extended to an edge consecutive matrix in the next definition.

Definition 3.19 For a multidigraph $G = (V, A, \psi)$, the edge consecutive matrix is the $l \times l$ matrix LJ with (a, b) entry

$$LJ(a,b) = \begin{cases} 1 & \text{if edges } a \text{ and } b \text{ are consecutive in order } ab \text{ in the graph } G, \\ 0 & \text{otherwise,} \end{cases} \tag{3.2}$$

where $a, b \in A$.

By definitions of the adjacency matrix and the line graph, the edge consecutive matrix is the adjacency matrix of the line graph of G. The directed graph shown in Fig. 3.7 is used as an example to construct the edge consecutive matrix as follows.

$$LJ = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}. \tag{3.3}$$

Definition 3.20 For a multidigraph $G = (V, A, \psi)$, the **incidence matrix** is the $m \times l$ matrix B with (v, a) entry

$$B(v, a) = \begin{cases} -1 & \text{if } a = (v, x) \text{ for some } x \in V, \\ 1 & \text{if } a = (x, v) \text{ for some } x \in V, \\ 0 & \text{otherwise,} \end{cases}$$

where $v \in V$ and $a \in A$.

According to the signed entries, the incidence matrix can be separated into the in-incidence matrix and the out-incidence matrix.

Definition 3.21 For a multidigraph $G = (V, A, \psi)$, the **in-incidence matrix** B_{in} and the **out-incidence matrix** B_{out} are the $m \times l$ matrices with (v, a) entries

$$B_{in}(v, a) = \begin{cases} 1 \text{ if } a = (x, v) \text{ for some } x \in V, \\ 0 \text{ otherwise,} \end{cases}$$

and

$$B_{out}(v, a) = \begin{cases} 1 \text{ if } a = (v, x) \text{ for some } x \in V, \\ 0 \text{ otherwise,} \end{cases}$$

where $v \in V$ and $a \in A$.

It is obvious that $B_{in} = (B + abs(B))/2$ and $B_{out} = (abs(B) - B)/2$, where $abs(B)$ denotes the matrix in which each entry equals the absolute value of the corresponding entry of B. For the directed graph shown in Fig. 3.7, the incidence matrix, the in-incidence matrix, and the out-incidence matrix are respectively expressed by

$$B = \begin{pmatrix} -1 & 0 & 1 & 0 & -1 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & 1 & -1 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 & -1 & 1 \\ 0 & 1 & 0 & -1 & 0 & 0 & 1 & -1 \end{pmatrix},$$

$$B_{in} = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}, \text{ and } B_{out} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}.$$

One finds that the incidence matrix depends on the label of each edge in a directed graph. To effectively analyze the graph model for conflict resolution using matrix representation, a unique rule of priority to label colored arcs is introduced in the next subsection.

Definition 3.22 A **colored multidigraph** (V, A, N, ψ, c) is a multidigraph (V, A, ψ) and a set of colors N, and a function $c : A \to N$ such that $c(a) \in N$ is the color of $a \in A$, provided that multiple edges of (V, A, ψ) are assigned different colors, i.e., if $a \neq b$, but $\psi(a) = \psi(b)$, then $c(a) \neq c(b)$.

If $a \in A$ such that $\psi(a) = (u, v)$ and $c(a) = i$ for $i \in N$, then a can be written as $a = d_i(u, v)$. The line digraph of $G = (V, A, N, \psi, c)$, $L(G)$, is a simple digraph and each vertex in $L(G)$ corresponds to an edge in the multidigraph G. Hence, coloring edges in G is equivalent to assigning colors to vertices in $L(G)$.

Definition 3.23 For a colored multidigraph (V, A, N, ψ, c), an **edge colored path** is a path in the multidigraph (V, A, ψ) in which each constituent edge has different colors.

If any two consecutive edges are restricted to having different colors in the edge consecutive matrix, this matrix is called the edge colored consecutive matrix. Its formal definition is as follows.

Definition 3.24 For a colored multidigraph $G = (V, A, N, \psi, c)$, the **edge colored consecutive matrix** LJ_c is the $l \times l$ matrix with (a, b) entry

$$LJ_c(a, b) = \begin{cases} 1 & \text{if edges } a \text{ and } b \text{ are consecutive in order } ab \\ & \text{and have different colors in the graph } G, \\ 0 & \text{otherwise.} \end{cases} \tag{3.4}$$

From algebraic graph theory (Godsil and Royle 2001), the following Lemma 3.1 that describes the relation between the adjacency matrix and incidence matrix can easily follow.

Lemma 3.1 *For a colored multidigraph $G = (V, A, N, \psi, c)$, the adjacency matrix J is expressed as*

$$J = sign[(B_{out}) \cdot (B_{in})^T]. \tag{3.5}$$

The following lemma that establishes the relation between the incidence matrix and the edge consecutive matrix is obtained based on Definition 3.21, on the in-incidence and out-incidence matrices B_{in} and B_{out}, and Definition 3.19, on the matrix LJ.

Lemma 3.2 *For a colored multidigraph $G = (V, A, N, \psi, c)$, B_{in} and B_{out} are the in-incidence matrix and out-incidence matrix of the graph G, respectively. Then, the edge consecutive matrix LJ satisfies $LJ = (B_{in})^T \cdot (B_{out})$.*

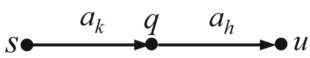


Fig. 3.8 a_k and a_h are consecutive in order $a_k a_h$

Proof Let $M = (B_{in})^T \cdot (B_{out})$. Any (k, h) entry of matrix M can be expressed as $M(k, h) = e_k^T \cdot M \cdot e_h = [(B_{in}) \cdot e_k]^T \cdot [(B_{out}) \cdot e_h]$, where e_k^T denotes the transpose of the kth standard basis vector of the l-dimensional Euclidean space.

Therefore, $M(k, h) \neq 0$ iff $B_{in}(q, a_k) \cdot B_{out}(q, a_h) \neq 0$ for some $q \in S$ such that $\psi(a_k) = (s, q)$ and $\psi(a_h) = (q, u)$ for $s, u \in S$. This implies that $M(a_k, a_h) \neq 0$ iff a_k and a_h are consecutive from a_k to a_h (See Fig. 3.8).

Hence, based on Definition 3.19, $M(a_k, a_h) \neq 0$ iff $LJ(a_k, a_h) \neq 0$. Since M and LJ are 0–1 matrices, then, $LJ = (B_{in})^T \cdot (B_{out})$ follows. □

3.3.2 A Rule of Priority to Label Colored Arcs

A colored multidigraph may contain several arcs with the same initial and terminal vertices, but each arc in this case must be assigned a different color (Xu et al. 2009b, 2013). To work with the set of all arcs, they must be carefully labeled. Assume that all colors and nodes are pre-numbered. Therefore, the vertex set V and the color set N in $G = (V, A, N, \psi, c)$ are numbered as $V = \{1, 2, \cdots, m\}$ and $N = \{1, 2, \cdots, n\}$, respectively. Let c_i denote the cardinality of arc set assigned color i, i.e., $c_i = |A_i|$, where $A_i = \{x \in A : c(x) = i\}$ for each $i \in N$.

To label the arcs in a colored multidigraph $G = (V, A, N, \psi, c)$, set $\varepsilon_0 = 0$ and $\varepsilon_i = \sum_{j=1}^{i} c_j$ for $i \in N$, and note that $l = \varepsilon_n = \sum_{i=1}^{n} c_i$ is the cardinality of A in G. The arcs, $a_1, a_2, \ldots, a_l$, will be labeled according to the color order; within each color, according to the sequence of initial nodes; and within each color and initial node, according to the sequence of terminal nodes. The ordering, referred to as the *Rule of Priority*, has the following properties:

1. If $\varepsilon_{i-1} < k \leq \varepsilon_i$, then $c(a_k) = i$, i.e., a_k has color i;
2. For $k < h$, if a_k and a_h both have color i for some $i \in N$, and if $\psi(a_k) = (v_x, v_y)$ and $\psi(a_h) = (v_z, v_w)$, then $x \leq z$ and, if $x = z$, then $y < w$.

If all arcs in a graph model have been labeled according to the Rule of Priority, then the index of an arc uniquely determines the DM controlling it. Therefore, $A_i = \{a_{\varepsilon_{i-1}+1}, \ldots, a_{\varepsilon_i}\}$, where A_i denotes the set of arcs with color i.

Recall that c_i denotes the cardinality of the arc set in color i and let E_{c_i} denote a $c_i \times c_i$ matrix with each entry being set to 1 for $i = 1, 2, \cdots, n$. Then, D is defined as the following block diagonal matrix

$$D = \begin{pmatrix} E_{c_1} & 0 & \cdots & 0 \\ 0 & E_{c_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & E_{c_n} \end{pmatrix}. \tag{3.6}$$

It is obvious that this matrix D encodes the color scheme in the graph G, where the dimension of each diagonal block E_{c_i} depends on the number of edges in color i. More specifically, recall that $\varepsilon_i = \sum_{j=1}^{i} c_j$ for $1 \leq i \leq n$. According to the Rule of Priority for labeling edges, for any $a_k \in A$ and $\varepsilon_{i-1} < k \leq \varepsilon_i$, the edge a_k has color i. Hence, for any $a_k, a_h \in A$, if there exists $1 \leq i \leq n$ such that $k, h \in (\varepsilon_{i-1}, \varepsilon_i]$, then edges a_k and a_h have the same color i, and $D(k, h) = 1$. Also, $D(k, h) = 0$ iff edges a_k and a_h have different colors.

This matrix captures the adjacency relation between pairs of consecutive edges without considering the color(s) of the consecutive edges. Another conversion function is thus presented next to transform the original problem of searching edge-colored paths in a colored multidigraph to the standard problem of finding paths in a simple digraph without color constraints. The conversion function can now be obtained in matrix form by the following lemma.

Lemma 3.3 *For a colored multidigraph $G = (V, A, N, \psi, c)$, let E_l be the $l \times l$ matrix with each entry equal to 1. Then the edge colored consecutive matrix LJ_c satisfies $LJ_c = LJ \circ (E_l - D)$, where "$\circ$" denotes the Hadamard product.*

Proof Let $LJ(k, h)$ and $(E_l - D)(k, h)$ denote the (k, h) entries of matrices LJ and $E_l - D$, respectively. Then, $LJ(k, h) \cdot (E_l - D)(k, h) \neq 0$ iff $LJ(k, h) \neq 0$ and $D(k, h) = 0$. Based on the definitions of matrices LJ and D, $LJ(k, h) \neq 0$ iff edges a_k and a_h are consecutive in order $a_k a_h$. $D(k, h) = 0$ iff edges a_k and a_h have different colors. Obviously, based on the definition of matrix LJ_c, $LJ_c = LJ \circ (E_l - D)$. □

Lemmas 3.2 and 3.3 together present a conversion function $F(B)$ such that

$$F(B) = [(B_{in})^T \cdot B_{out}] \circ (E_l - D), \tag{3.7}$$

where $B_{in} = (B + abs(B))/2$ and $B_{out} = (abs(B) - B)/2$. Therefore, $F(B)$ transforms a problem of searching colored paths in an edge colored digraph to a standard problem of finding paths in a simple digraph with no color constraints. Note that the incident relations between vertices and edges of a graph can uniquely characterize the graph. Therefore, the incidence matrix is treated as the original graph and used for computer implementation.

Example 3.4 (*Rule of Priority and Edge Colored Consecutive Matrix.*) A sustainable development game to model a conflict between an environmental agency and a developer was considered by Hipel (2001) and Li et al. (2004). The conflict is

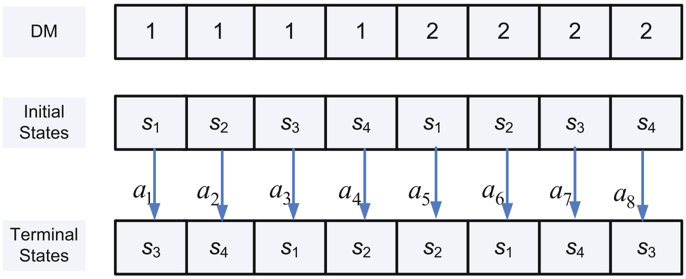


Fig. 3.9 The labels of edges

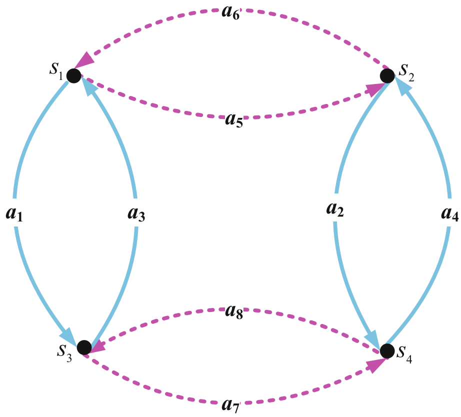


Fig. 3.10 Labeled graph model for the sustainable development conflict

modeled by two DMs: an environmental agency (DM 1) and a developer (DM 2). The graph model $G = (S, A, c)$ for the sustainable development conflict is depicted in Fig. 3.2, where vertices designate states and arcs represent movement between states. The number on a given arc indicates which DM controls the movement while the arrowhead shows the direction of movement. According to the Rule of Priority, label the edges of the graph model $G = (S, A, c)$ and calculate its edge colored consecutive matrix.

Assume that the DM set $N = \{1, 2\}$ and state set $S = \{s_1, s_2, s_3, s_4\}$. The cardinalities of the arc sets A_1 and A_2 are 4, respectively. Then, according to the Rule of Priority, the oriented arcs are numbered as in Fig. 3.9. The sustainable development game is expressed as the labeled graph model presented in Fig. 3.10 in which the full curves and dotted curves denote DM 1 and DM 2, respectively. Specifically, $a_1 = (s_1, s_3)$ and $c(a_1) = 1$; $a_2 = (s_2, s_4)$ and $c(a_2) = 1$; $a_3 = (s_3, s_1)$ and $c(a_3) = 1$; $a_4 = (s_4, s_2)$ and $c(a_4) = 1$; $a_5 = (s_1, s_2)$ and $c(a_5) = 2$; $a_6 = (s_2, s_1)$ and $c(a_6) = 2$; $a_7 = (s_3, s_4)$ and $c(a_7) = 2$; and $a_8 = (s_4, s_3)$ and $c(a_8) = 2$. Therefore, $A_1 = \{a_1, a_2, a_3, a_4\}$ and $A_2 = \{a_5, a_6, a_7, a_8\}$. Since,

$$E_l - D = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \end{pmatrix},$$

and LJ given in Eq. 3.3, one can obtain

$$LJ_c = LJ \circ (E_l - D) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

3.3.3 Adjacency Matrix and Reachable List

Important matrices associated with a digraph include the adjacency matrix and the incidence matrix (Godsil and Royle 2001). Let $m = |V|$ denote the number of vertices and $l = |A|$ be the number of edges of the directed graph G. Then,

Definition 3.25 For a graph model $G = (S, A)$, DM i's **adjacency matrix** is the $m \times m$ matrix J_i with (s, q) entry

$$J_i(s, q) = \begin{cases} 1 & \text{if } (s, q) \in A_i, \\ 0 & \text{otherwise,} \end{cases}$$

where $s, q \in S$.

Let $i \in N$ and $s \in S$. $R_i(s)$ denotes DM $i's$ reachable list from a state s, containing all states to which DM i can move from state s in one step. $R_i(s)$ represents DM $i's$ unilateral moves (UMs). If $R_i(s)$ is written as a 0–1 row vector, then DM i's adjacency matrix J_i and reachable list from state s have the relation

$$R_i(s) = e_s^T \cdot J_i,$$

where e_s^T denotes the transpose of the sth standard basis vector of the m-dimensional Euclidean space.

For the graph model of the sustainable development game presented in Fig. 3.2, the adjacency matrices for DM 1 and DM 2 are

$$J_1 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \text{ and } J_2 = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

3.3.4 Preference Matrices

Preference information plays an important role in a graph model. A set of preference matrices can represent preference relations between any two states with different requirements. Two $m \times m$ preference matrices for DM i in the graph model with simple preference are defined as follows:

$$P_i^+(s, q) = \begin{cases} 1 & \text{if } q \succ_i s, \\ 0 & \text{otherwise,} \end{cases} \tag{3.8}$$

and

$$P_i^{-,=}(s, q) = \begin{cases} 1 & \text{if } s \succ_i q \text{ or } (s \sim_i q \text{ and } s \neq q), \\ 0 & \text{otherwise.} \end{cases} \tag{3.9}$$

The preference matrix P_i^+ may be used to represent more preferred relations and the preference matrix $P_i^{-,=}$ can represent less preferred or equally preferred relations between any two states. Since simple preference structure is complete, matrices P_i^+ and $P_i^{-,=}$ have the relation $P_i^+ = E - I - P_i^{-,=}$, where E is the $m \times m$ matrix with each entry equal to 1 and I the $m \times m$ identity matrix. For example, the sustainable development model provides the preference information for two DMs as follows:

$$s_1 \succ_1 s_3 \succ_1 s_2 \succ_1 s_4 \text{ and } s_3 \succ_2 s_1 \succ_2 s_4 \succ_2 s_2.$$

Therefore, two DMs' preference matrices are expressed by

$$P_1^+ = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix}, P_1^{-,=} = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix},$$

$$P_2^+ = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}, \text{ and } P_2^{-,=} = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}.$$

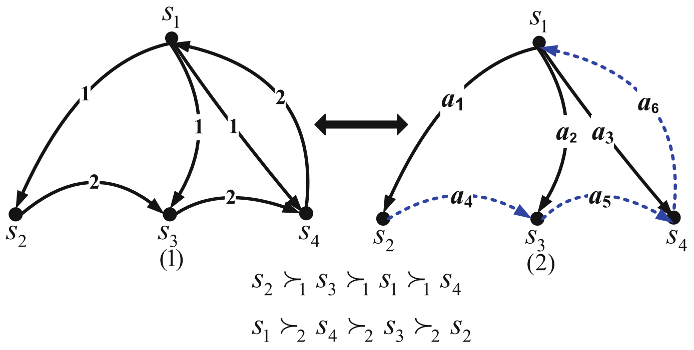


Fig. 3.11 Different representations of a graph model

3.3.5 Incidence Matrix and Graph Model

The incidence matrix based on the Rule of Priority with preference matrices can completely represent a graph model. Figure 3.11 depicts a graph model and its edge labeled graph. Although no DM is explicitly shown in the labeled graph, the index number of an arc uniquely determines the DM who controls it when all arcs have been numbered according to the Rule of Priority. Specifically, based on the number of arcs in i's graph G_i for $i = 1, 2$, $|A_1| = 3$, and $|A_2| = 3$, arcs a_1 to a_3 are controlled by DM 1, arcs a_4 to a_6 by DM 2.

The incidence matrix of the labeled graph

$$B = \begin{pmatrix} -1 & -1 & -1 & 0 & 0 & 1 \\ 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 & 1 & -1 \end{pmatrix},$$

and the preference matrices base on DMs' preference information

$$P_1^+ = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix}, \text{ and } P_2^+ = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \end{pmatrix},$$

can represent the graph model shown in Fig. 3.11.

3.4 Important Ideas

Conflicts arise across a wide range of scales and settings. To model a strategic conflict, the normal form, option form, graph model form and its matrix representation are introduced in this chapter. Compared with the normal and option forms to represent strategic conflicts, the graph model has several advantages, including its ability to

- handle irreversible moves,
- model common moves,
- provide a flexible framework for defining, comparing, and characterizing solution concepts, and
- be easily applied to actual conflicts.

For small or generic conflicts, the normal form of the game explained in Sect. 3.1.1 can be a convenient notation to employ, as is shown in Table 3.2 for the 2×2 sustainable development game. In practice, the option form of the game defined in Sect. 3.1.2 and illustrated in Table 3.3 for the sustainable development conflict constitutes a flexible format to use in practice for recording conflicts ranging from simple to complicated ones. In fact, option form is utilized in the vast majority of cases for defining states needed in the graph model formulation. After a graph model is converted to a labeled digraph based on the proposed Rule of Priority, it can be represented by using a set of matrices that can be utilized to analyze a graph model using algebraic graph theory. In the next chapter, stability definitions (or solution concepts) are defined logically, in terms of the underlying graphs, and formulated explicitly using matrices for the case of what is called simple preference.

3.5 Problems

3.5.1 The normal form of the game is displayed in Table 3.2 for the sustainable development game. Because each of the two DMs controls two strategies, this is called a 2×2 game. These small 2×2 games represent the simplest possible game that could occur and can be highly informative for clearly explaining the strategic interpretation of conflict situations that can arise in the real-world, such as the sustainable development game. A widely known 2×2 game is called Prisoner's Dilemma which is used to reflect the situation in which a DM must decide whether to act in his or her own interest in the short term or to cooperate with another DM, in order to reach a better result in the longer term. The 2×2 normal form of this conflict is written as given in Table 3.6.

In Prisoner's Dilemma, notice that if the two decision makers labeled as DM 1 and DM 2 cooperate with one another, they both fare reasonable well (state s_1) compared to the situation in which they do not (state s_4).

(a) By referring to a well known book or paper on 2×2 games, explain in English what conflict is taking place between the two prisoners.

Table 3.6 Prisoners Dilemma in normal form

		DM 2	
		Cooperate	Do Not Cooperate
DM 1	Cooperate	3, 3 s_1	1, 4 s_2
	Do Not Cooperate	4, 1 s_3	2, 2 s_4

Table 3.7 The game of Chicken in normal form

		DM 2	
		Do Not Swerve	Swerve
DM 1	Do Not Swerve	1, 1 s_1	4, 2 s_2
	Swerve	2, 4 s_3	3, 3 s_4

(b) Using a real-world example, explain how a situation involving labour and management could be reasonably modeled using Prisoner's Dilemma.
(c) Describe how and why the conflict over climate change could be interpreted in its simplest form using Prisoner's Dilemma.

3.5.2 For the Prisoner's Dilemma game mentioned in Problem 3.5.1:

(a) Record the option form of this conflict.
(b) Show the graph model version of this dispute.

3.5.3 In repeated Prisoner's Dilemma, the two competitors deal with each other on a regular basis over time. By referring to the literature, explain the best strategy to follow in repeated Prisoner's Dilemma.

3.5.4 The famous game of Chicken is another well known 2 × 2 game which can be written in normal form as shown in Table 3.7.

In this high risk confrontation, two drivers, called DM 1 and DM 2, are driving at high speed towards one another. The driver, who swerves off the road to avoid a collision in which both drivers would be killed, loses the game and is called a chicken. Notice in the game of Chicken that the worst situation is when both drivers do not swerve, which is state s_1.

(a) Explain why the preferences for each DM in Chicken make sense.
(b) The Cuban Missile Crisis of 1962 is sometimes modeled as a game of Chicken. By locating appropriate references, outline what happened in the Cuban Missile Crisis. Write down the Cuban Missile Crisis in normal form as a game of Chicken.
(c) Describe another situation involving the game of Chicken which could take place in the real-world.

3.5.5 In Problem 3.5.4, it is mentioned that the Cuban Missile Crisis is sometimes interpreted as a game of Chicken. Rather than using the game of Chicken, Fraser and Hipel (1984, Chap. 2) and also Hipel (2011) develop a much more realistic model of the Cuban Missile Crisis in option form.

(a) Show the normal form of the Cuban Missile Crisis mentioned above.
(b) Write down the option form of the Cuban Missile Crisis.
(c) Show the graph model for the Cuban Missile Crisis.

3.5.6 Sometimes misunderstandings can arise in a conflict situation, which is referred to as a hypergame as mentioned in Sect. 10.3.1. By referring to the literature, qualitatively explain what is meant by a hypergame. Explain why the Cuban Missile Crisis would be best modeled as a hypergame.

3.5.7 Write down the matrix or algebraic form of the Prisoner's Dilemma game mentioned in Problem 3.5.1.

3.5.8 Show the matrix or algebraic formulation of the game of Chicken mentioned in Problem 3.5.4.

3.5.9 For the graph model shown in Fig. 3.11,

(a) label the graph model (1) to present all processes according to the Rule of Priority;
(b) calculate its edge consecutive matrix and edge colored consecutive matrix.

3.5.10 A superpower nuclear confrontation (Fang et al. 1993) is modeled using two DMs and six options. These options determine five feasible states as listed in Table 3.8. Note that state W represents a nuclear winter. The graph model is shown in Fig. 3.12.

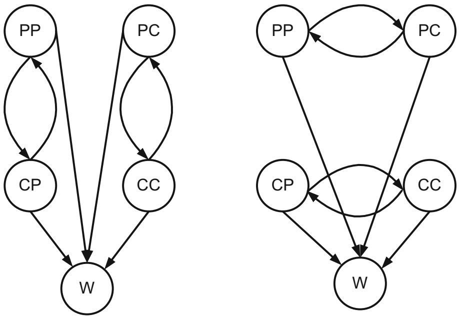
For the graph model shown in Fig. 3.12:

(a) label the graph model to present all processes according to the Rule of Priority;
(b) calculate its incidence matrix and edge colored consecutive matrix.

Table 3.8 Decision makers, options and feasible states for the superpower nuclear confrontation conflict

DM 1					
1. Peace (P)	Y	Y	N	N	N
2. Conventional attack (C)	N	N	Y	Y	N
3. Full nuclear attack (W)	N	N	N	N	Y
DM 2					
1. Peace (P)	Y	N	Y	N	N
2. Conventional attack (C)	N	Y	N	Y	N
3. Full nuclear attack (W)	N	N	N	N	Y
States	**PP**	**PC**	**CP**	**CC**	**W**

Fig. 3.12 The graph model of the superpower nuclear confrontation conflict



(a) Graph model for DM 1 (b) Graph model for DM 2

DM 1: $PP \succ_1 CP \succ_1 CC \succ_1 PC \succ_1 W$
DM 2: $PP \succ_2 PC \succ_2 CC \succ_2 CP \succ_2 W$

References

Bernath Walker, S., Hipel, K. W., & Xu, H. (2013). A matrix representation of attitudes in conflicts. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *43*(6), 1328–1342. https://doi.org/10.1109/tsmc.2013.2260536.

Buckley, F., & Harary, F. (1990). *Distance in graphs*. Reading: Addison-Wesley.

Dieste, R. (1997). *Graph theory*. New York: Springer. https://doi.org/10.1007/978-1-4419-1153-7_402.

Fang, L., Hipel, K. W., & Kilgour, D. M. (1993). *Interactive decision making: The graph model for conflict resolution*. New York: Wiley. https://doi.org/10.2307/2583940.

Fraser, N. M., & Hipel, K. W. (1984). *Conflict analysis: Models and resolutions*. New York: North-Holland. https://doi.org/10.2307/2582031.

Godsil, C., & Royle, G. (2001). *Algebraic graph theory*. New York: Springer. https://doi.org/10.1007/978-1-4613-0163-9.

Gondran, M., & Minoux, M. (1979). *Graphs and algorithms*. New York: Wiley.

Hipel, K. W. (2001). Conflict resolution. In M. K. Tolba (Ed.), *Our fragile world: Challenges and opportunities for sustainable development (Forerunner to the Encyclopedia of Life Support Systems)* (Vol. 1, pp. 935–952). Oxford, United Kingdom: Eolss.

Hipel, K. W. (2011). A systems engineering approach to conflict resolution in command and control. *The International C2 Journal*, *5*(1), 1–56.
Hoffman, A. J., & Schiebe, B. (2001). The edge versus path incidence matrix of series-parallel graphs and greedy packing. *Discrete Applied Mathematics*, *113*, 275–284. https://doi.org/10.1016/s0166-218x(00)00294-8.
Hou, Y., Jiang, Y., & Xu, H. (2015). Option prioritization for three-level preference in the graph model for conflict resolution. In B. Kaminski, G. E. Kersten & T. Szapiro (Eds.), *Outlooks and insights on group decision and negotiation, the Proceedings of the 2015 International Conference on Group Decision and Negotiation* (pp. 269–280). Cham, Switzerland: Springer.
Howard, N. (1971). *Paradoxes of rationality: Theory of metagames and political behavior*. Cambridge: MIT Press. https://doi.org/10.2307/1266876.
Li, K. W., Hipel, K. W., Kilgour, D. M., & Fang, L. (2004). Preference uncertainty in the graph model for conflict resolution. *IEEE Transactions on Systems, Man, and Cybernetics, Part A, Systems and Humans*, *34*(4), 507–520. https://doi.org/10.1109/tsmca.2004.826282.
Shiny, A. K., & Pujari, A. K. (1998). Computation of prime implicants using matrix and paths. *Journal of Logic and Computation*, *8*(2), 135–145. https://doi.org/10.1093/logcom/8.2.135.
Von Neumann, J., & Morgenstern, O. (1953). *Theory of games and economic behavior* (3rd ed.). Princeton: Princeton University Press.
Xu, H., Kilgour, D. M., & Hipel, K. W. (2007). Matrix representation of solution concepts in graph models for two decision-makers with preference uncertainty. *Dynamics of Continuous, Discrete and Impulsive Systems*, *14*(S1), 703–707.
Xu, H., Hipel, K. W., & Kilgour, D. M. (2009a). Matrix representation of solution concepts in multiple decision maker graph models. *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, *39*(1), 96–108. https://doi.org/10.1109/tsmca.2009.2007994.
Xu, H., Li, K. W., Hipel, K. W., & Kilgour, D. M. (2009b). A matrix approach to status quo analysis in the graph model for conflict resolution. *Applied Mathematics and Computation*, *212*(2), 470–480. https://doi.org/10.1016/j.amc.2009.02.051.
Xu, H., Li, K. W., Kilgour, D. M., & Hipel, K. W. (2009c). A matrix-based approach to searching colored paths in a weighted colored multidigraph. *Applied Mathematics and Computation*, *215*, 353–366. https://doi.org/10.1016/j.amc.2009.04.086.
Xu, H., Hipel, K. W., Kilgour, D. M., & Chen, Y. (2010a). Combining strength and uncertainty for preferences in the graph model for conflict resolution with multiple decision makers. *Theory and Decision*, *69*(4), 497–521. https://doi.org/10.1007/s11238-009-9134-6.
Xu, H., Kilgour, D. M., & Hipel, K. W. (2010b). An integrated algebraic approach to conflict resolution with three-level preference. *Applied Mathematics and Computation*, *216*(3), 693–707. https://doi.org/10.1016/j.amc.2010.01.054.
Xu, H., Kilgour, D. M., & Hipel, K. W. (2010c). Matrix representation and extension of coalition analysis in group decision support. *Computers and Mathematics with Applications*, *60*(5), 1164–1176. https://doi.org/10.1016/j.camwa.2010.05.040.
Xu, H., Kilgour, D. M., Hipel, K. W., & Kemkes, G. (2010d). Using matrices to link conflict evolution and resolution in a graph model. *European Journal of Operational Research*, *207*, 318–329. https://doi.org/10.1016/j.ejor.2010.03.025.
Xu, H., Kilgour, D. M., & Hipel, K. W. (2011). Matrix representation of conflict resolution in multiple-decision-maker graph models with preference uncertainty. *Group Decision and Negotiation*, *20*(6), 755–779. https://doi.org/10.1007/s10726-010-9188-4.
Xu, H., Kilgour, D. M., Hipel, K. W., & McBean, E. A. (2013). Theory and application of conflict resolution with hybrid preference in colored graphs. *Applied Mathematical Modelling*, *37*(3), 989–1003. https://doi.org/10.1016/j.apm.2012.03.009.
Xu, H., Kilgour, D. M., Hipel, K. W., & McBean, E. A. (2014). Theory and implementation of coalitional analysis in cooperative decision making. *Theory and Decision*, *76*(2), 147–171. https://doi.org/10.1007/s11238-013-9363-6.