Elements of Network

Project Management and Monitoring

Characteristics of CPM/PERT Projects

Here are some characteristics of CPM/PERT projects:

1. **Large Scale:** CPM/PERT projects are typically large and complex, with many activities and interdependent tasks that require coordination and tracking.
2. **Time Sensitive:** These projects are often time-sensitive, with deadlines that need to be met. Effective planning and scheduling are crucial to ensure that the project is completed on time.
3. **Complex Activities:** The activities involved in these projects are often complex and require specific skills and expertise. Multiple teams may be involved, each responsible for different tasks.

4. **Interdependent tasks:** The activities are often interdependent, meaning that one task cannot be completed until another task is finished. The order and timing of activities are critical to the success of the project.
5. **Uncertainty and Risk:** CPM/PERT projects involve uncertainty and risk. The schedule may be affected by unexpected events, such as delays, equipment failure, or unforeseen changes in scope.
6. **Resource Management:** Resource management is a critical aspect of CPM/PERT projects. Effective allocation of resources, including time, money, and people, is essential to ensure that the project is completed on time and within budget.

7. **Critical Path:** The critical path is the sequence of activities that determines the duration of the project. The critical path activities are the ones that must be completed on time to ensure the project is delivered within the specified time frame.
8. **Monitoring and Control:** CPM/PERT projects require continuous monitoring and control to ensure that the project stays on track. Regular review and updates to the schedule are necessary to identify potential delays or problems and take corrective action.

Events in PERT/CPM

1. Represent milestones: Events represent important milestones in the project schedule, such as the start or completion of an activity.
2. **Have a specific time:** Each event has a specific time associated with it, which is either the start or finish time of an activity.
3. **Have a unique identifier:** Each event is given a unique identifier or number to distinguish it from other events in the diagram.

4. **Can be predecessors or successors:** Events can be predecessors or successors to other events, depending on their relationship with other activities in the project.
5. **Are interconnected:** Events are connected by arrows, which represent the activities that link them together. The arrows indicate the sequence in which the activities must be completed.
6. **Can have multiple predecessors or successors:** Some events can have multiple predecessors or successors, depending on the complexity of the project.

Representation of Events

Events are represented by nodes in a network. It may have any of the following shapes.

(i) Circular *(ii) Square* *(iii) Rectangular* *(iv) Oval*

Fig. *Different Shapes for Events*

Specifying the Events

A particular event out of various events on the network diagram may be specified as:

- Tail Event
- Head Event
- Dual Role Event

Tail Event

A tail event or a start event of a project: It has only outgoing arrows.

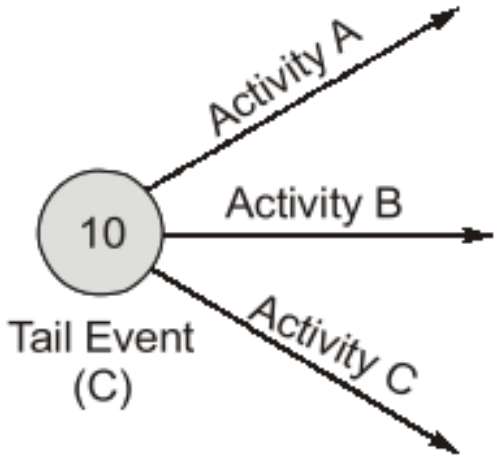


Fig. *Tail Event*

Event 10 is a tail event. Arrows represent job or activity of a project.

Head Event

Head event or final event: It is finish of a project having only incoming arrows.

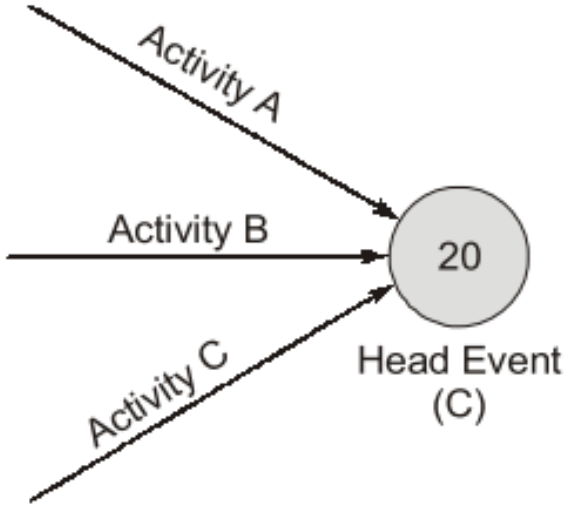


Fig. *Head Event*

Event 20 is a Head Event

Dual Role Event

Dual role events: All events except the first and the last event of a project are dual role events. They have both incoming and outgoing arrows.

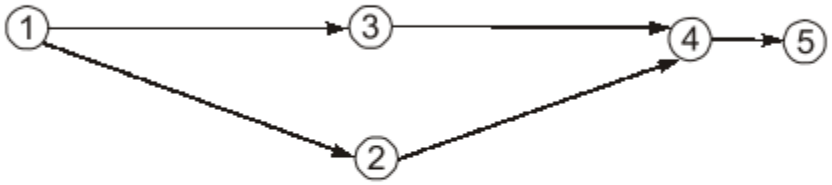


Fig. *Dual Role Events*

e.g.: Events 2, 3 and 4, are dual role events.

Events 2, 3, 4, are dual role events

Events 2, 3, 4, are dual role events

Successor events : The event or events that follow another event are called successor events to that event.

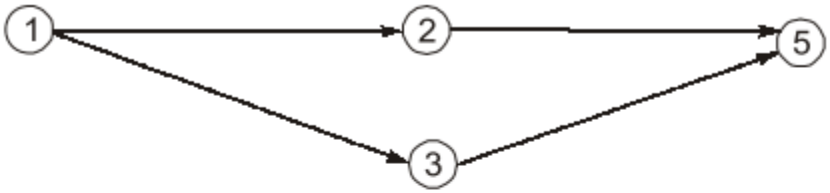


Fig. *Successor Events*

e.g.: Event 2 and 3 are successor events of event 1.

Event 2, 3 are successor events of 1

Predecessor events : The event or events that occur before another event are called predecessor event to that event.

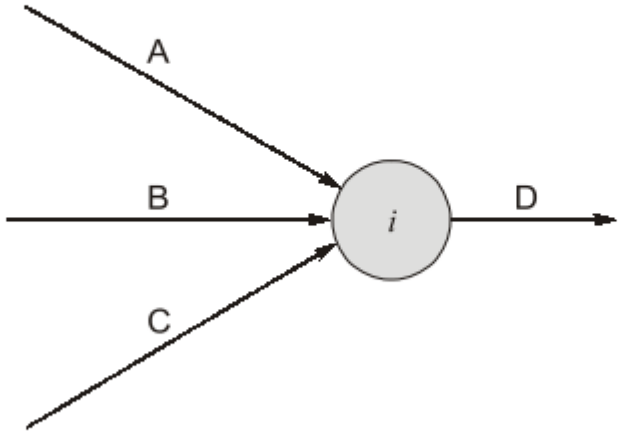
In above figure, events 2, 3 are predecessor to event 5.

NOTE: The project should have only one Tail event and Head event

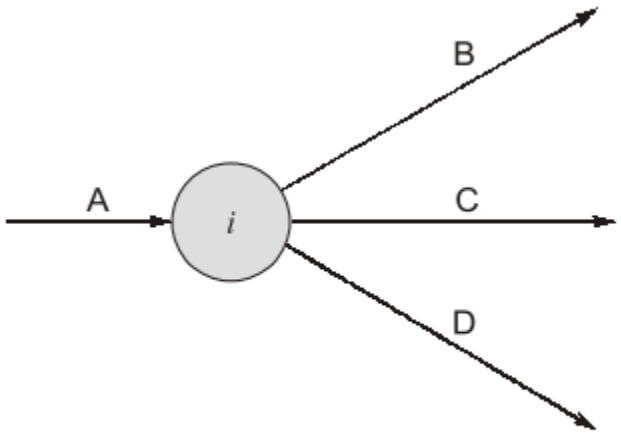
Types of Activities

1. **Parallel activities:** Parallel activities are those which can exist simultaneously or concurrently and are independent of each other.

   Ex:

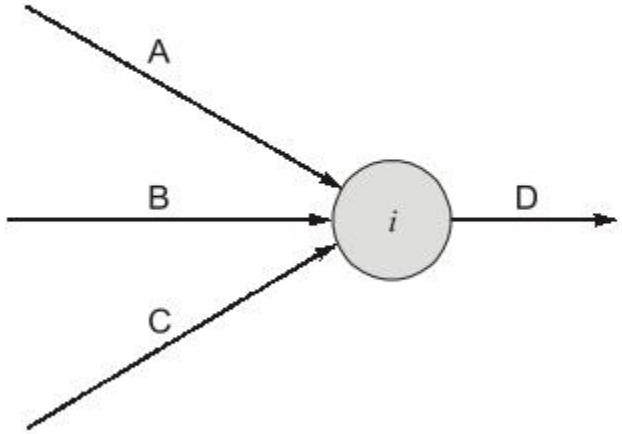


A, B, C are concurrent activities and terminates at same node '*i*' such activities are independent of each other.



As soon as event '*i*' reached, activity B, C and D can be simultaneously started. Such activities (B, C, D) are also parallel activities.

2. **Serial activities:** The activities which can be started one after another and are dependent on each other are called as serial activities.



The set of activities A, B and C are in series with activity D.

3. **Successor and precedessor activity:** The activity which exist after the occurrence of other activity is referred as successor activity. If this activity exist immediately after the occurrence of other activities it is referred as immediate successor activity.

The activity which exist before the occurrence of other activity is referred as precedessor activity. If this activity exist immediately before the occurrence of the other activities is referred as immediate precedessor activity.

Dummy

- A dummy is a type of operation which neither requires time nor any resource, but it denotes dependency among the activities.
- It is represented by dashed arrow.

  In the figure shown below, a dummy activity is shown.

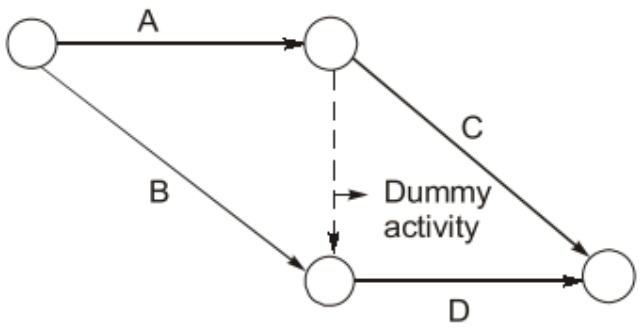


Fig. *Dummy Activity*

Activity

An activity refers to a specific task or work item that needs to be completed as part of a project. Activities are usually defined in the project schedule and are performed sequentially or in parallel with other activities.

- Each activity has a unique set of characteristics that include a start and end date, a duration, and dependencies on other activities.
- Activities can be broken down into smaller components known as work packages, which are easier to manage and track progress against.

Activities are critical to the success of a project because they define the work that needs to be done, who is responsible for doing it, and when it needs to be completed. They also help project managers allocate resources, track progress, and identify any potential delays or issues that may arise.

Representation and Identification of an Activity

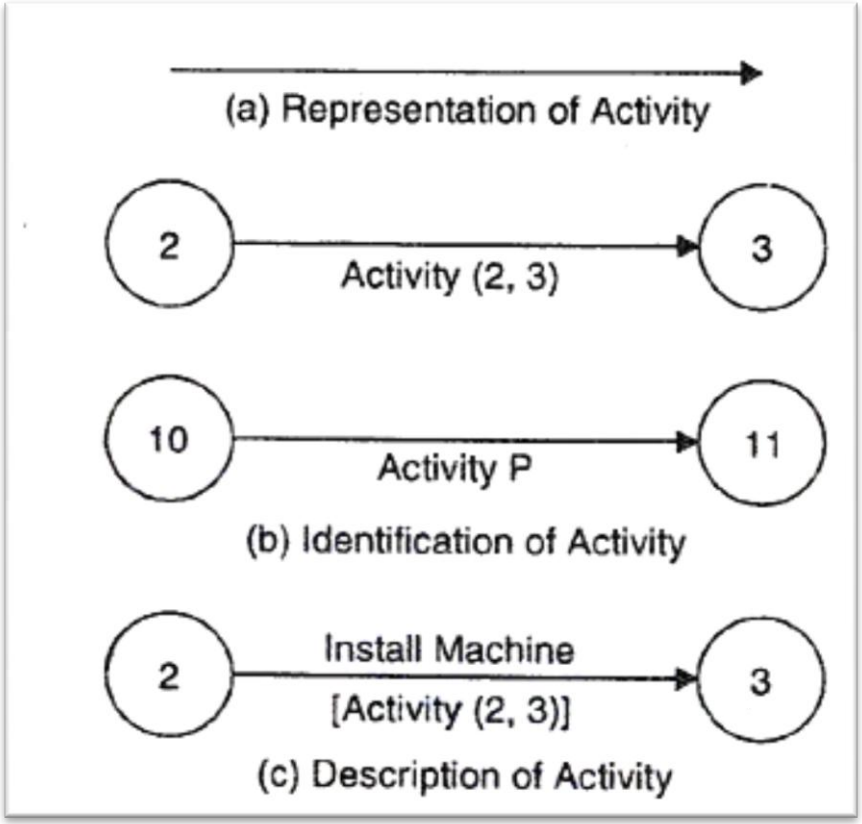
One common way to represent and identify an activity in project management is by using a task list or a Gantt chart.

A task list is a simple list of all the activities that need to be completed as part of the project, along with their start and end dates, duration, and any dependencies they may have on other activities. For example:

Task	Start Date	End Date	Duration	Dependencies
Design website layout	1/1/2023	1/7/2023	6 days	None
Develop website content	1/8/2023	1/21/2023	14 days	Design website layout
Test website functionality	1/22/2023	2/4/2023	14 days	Develop website content

- the other hand, is a graphical representation of the project schedule that shows each activity as a horizontal bar spanning the duration of the activity. Dependencies between activities are represented as arrows connecting the bars. The Gantt chart below shows the same activities as the task list above:
- In this example, the design website layout activity starts on January 1, 2023, and ends on January 7, 2023. It has no dependencies on other activities, so it can start as soon as the project begins. The develop website content activity cannot start until the design website layout activity is completed, so it has a dependency on that activity. Similarly, the test website functionality activity cannot start until the develop website content activity is completed.
- Representing and identifying activities in this way allows project managers to easily see the overall project schedule, track progress, and identify any potential delays or issues that may arise.

In a network diagram, activities are represented by simple arrows, usually drawn from left to right. The length of arrow does neither represent the magnitude of work involved nor the time required for its completion. It is thus not a vector quantity. The length of the arrow is chosen to suit the drafting convenience.



Representing and identifying activities in this way allows project managers to easily see the overall project schedule, track progress, and identify any potential delays or issues that may arise.

Inter-relationships of Activities

Inter-relationships of activities in project management refer to the dependencies between different project activities, and how they impact the overall project schedule. There are four types of inter-relationships between activities:

- **Finish-to-Start (FS):** This is the most common type of inter-relationship, where one activity cannot start until another activity has finished. For example, the installation of a new software application cannot start until the development of the software has been completed.

- **Start-to-Start (SS):** This type of inter-relationship means that one activity cannot start until another activity has started. For example, the testing of a software application cannot start until the development of the software has begun.
- **Finish-to-Finish (FF):** This type of inter-relationship means that one activity cannot finish until another activity has finished. For example, the training of staff cannot finish until the installation of new equipment has been completed.
- **Start-to-Finish (SF):** This is the least common type of inter-relationship, where one activity cannot finish until another activity has started. For example, the recruitment of new staff cannot finish until the training of existing staff has start

It's important for project managers to understand the inter-relationships between activities to ensure that the project schedule is realistic and achievable.

Project managers can use project management tools such as network diagrams or Gantt charts to visualize these inter-relationships and make adjustments to the project schedule as needed.

By managing the inter-relationships between activities effectively, project managers can ensure that the project is completed on time and within budget.

Depending up on the Dependencies Activities can be Categorised as

- Parallel Activities
- Serial Activities

Parallel Activities

Parallel activities in project management refer to activities that can be performed simultaneously rather than in a sequential order. By performing parallel activities, project managers can accelerate the project schedule and reduce the overall project duration. Here are some examples of parallel activities in project management:

- **Design and Procurement:** In a construction project, while the architectural and engineering design work is being done, procurement activities such as obtaining permits, ordering materials, and equipment can be started at the same time. This can help to speed up the overall project schedule.
- **Coding and Testing:** In a software development project, coding and testing activities can be done in parallel. While the coding is underway, testing can begin on the code that has already been written. This can help to identify and resolve issues early on in the development process.

- **Construction and Commissioning:** In a manufacturing plant construction project, construction activities can be done in parallel with commissioning activities. As the construction work progresses, commissioning activities such as testing and calibrating equipment can begin.
- **Marketing and Production:** In a new product development project, marketing activities such as market research, advertising, and branding can be done in parallel with production activities. This can help to ensure that there is a market for the product when it is launched, and that the production process is optimized for efficiency.

Serial Activity

Serial activities in project management refer to activities that must be performed in a specific sequence or order, one after the other. These activities cannot start until the previous activity is completed. Here are some examples of serial activities in project management:

1. **Requirements Gathering and Design:** In a software development project, requirements gathering activities such as stakeholder interviews and data analysis must be completed before the design work can begin. This is because the design work is based on the requirements gathered during this phase.

2. **Pouring Foundation and Framing:** In a construction project, pouring the foundation must be completed before the framing work can begin. This is because the framing is supported by the foundation, and the size and location of the foundation must be determined before the framing can be designed and installed.
3. **Product Design and Tooling:** In a manufacturing project, the product design must be completed before the tooling can be developed. This is because the tooling is used to manufacture the product, and it must be designed and developed based on the product specifications.

4. **Market Research and Product Development:** In a new product development project, market research must be completed before product development can begin. This is because the product development process must take into account customer needs and preferences, which are determined through market research.

Serial activities are important in project management because they establish the necessary sequence for completing the project work. By identifying serial activities and ensuring that they are completed in the correct order, project managers can optimize the project schedule and avoid delays or rework. However, it's important to ensure that the project schedule allows enough time for each serial activity to be completed before the next one begins, and that dependencies between activities are clearly understood and managed.

Classification of Activities

- Successor Activities
- Predecessor Activities
- Dummy Activities

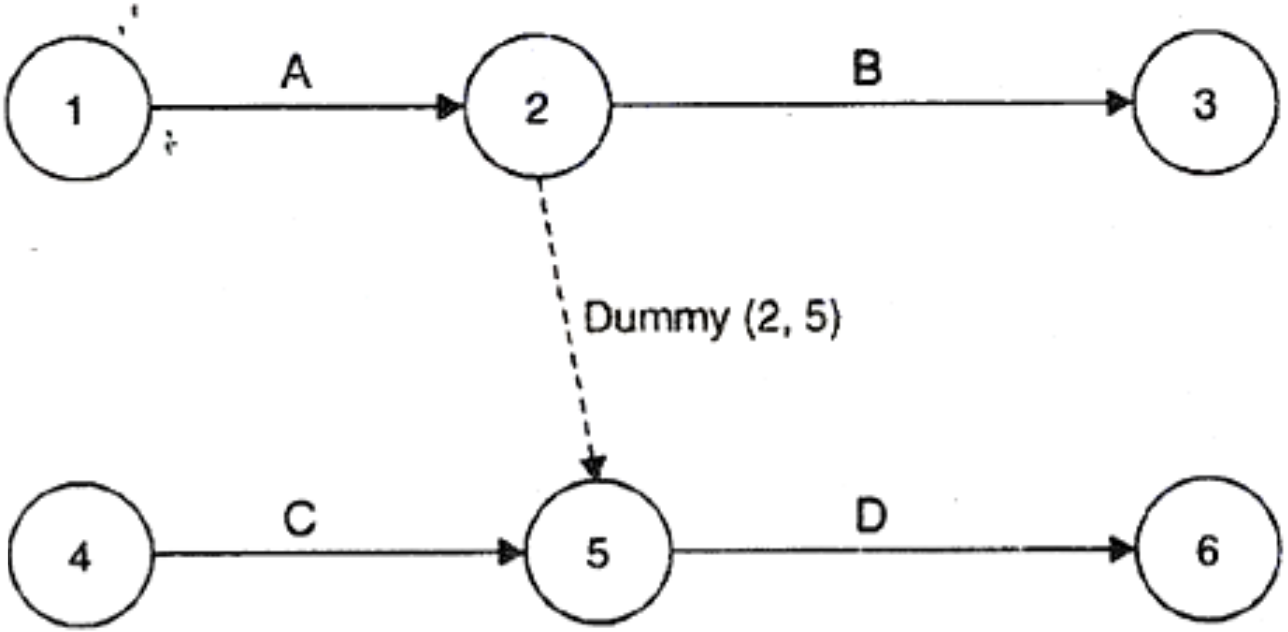
Dummy Activity

A dummy activity in project management is a type of activity that is added to a project network diagram to show dependencies between activities. A dummy activity has zero duration and no resources are assigned to it. It is represented by a dotted line with an arrowhead and usually labelled with a letter "D" or "dummy".

The purpose of a dummy activity is to show the logical relationship between two activities that have a dependency, but no physical relationship. For example, in a construction project, the installation of drywall cannot begin until the installation of electrical wiring is completed. However, there is no physical connection between these two activities, so a dummy activity is added to the project network diagram to show the logical dependency between them.

Dummy activities are also used when two or more activities have the same starting and ending nodes. In this case, a dummy activity is added to differentiate between the activities and show their order of execution.

It's important to note that dummy activities do not add any value to the project and do not consume any resources. They are used only for the purpose of visualizing the logical dependencies between activities in the project network diagram. However, they can be helpful in identifying potential scheduling conflicts and ensuring that the project schedule is realistic and achievable.



DUMMY OPERATION.

Set 1. *A*. A wait delivery of new machine.

B. Install new machine.

Set 2. *C*. Remove existing machine.

D. Dispose of existing machine.

Activities *A* and *B* are to be performed serially. Similarly, activities *C* and *D* are to be performed serially. Both the sets are performed simultaneously. However, from practical considerations,

we find that activity D of set 2 cannot be performed unless activity A of set 1 is completed. Hence a *dummy link* is used, joining node 2 to node 5, indicating that activity D cannot be started unless event 2 is over.

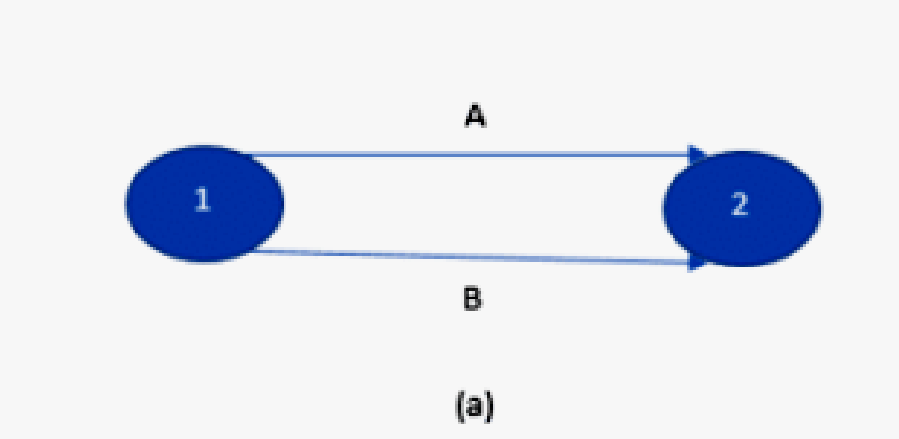
Use of Dummy Activity

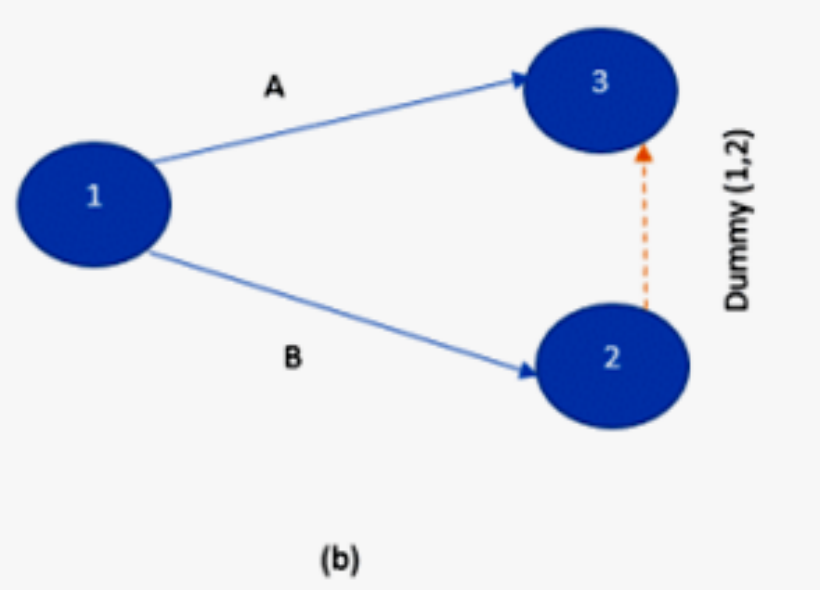
The dummy activity serves the following purposes in a project network:

- Grammatical Purpose
- Logical Purpose

1. Grammatical Purpose

A dummy activity can be used to prevent two arrows with a common beginning and end points. This can be explained by an example as shown in the figure.





Figures: (a)Network without Dummy Activity
(b)Dummy Activity Serving the Grammatical Purpose

Consider the arrows of activities A and B. Both start from node 1 and end at node 2. This arrangement is difficult to conduct computations and the network loses its uniqueness in its identification. Such inconvenience causes frequent mistakes during network analysis.

2. Logical Purpose

It is difficult to represent an activity having two sets of operation running parallel to each other in a network. The use of a dummy activity helps to give a logical representation without difficulty in interpretation.

For example, consider the network given in figure, where there are two activities Q and R. Both have a common end node. The predecessors of Q are **O** and **P** while that of R are **P** and **N**.

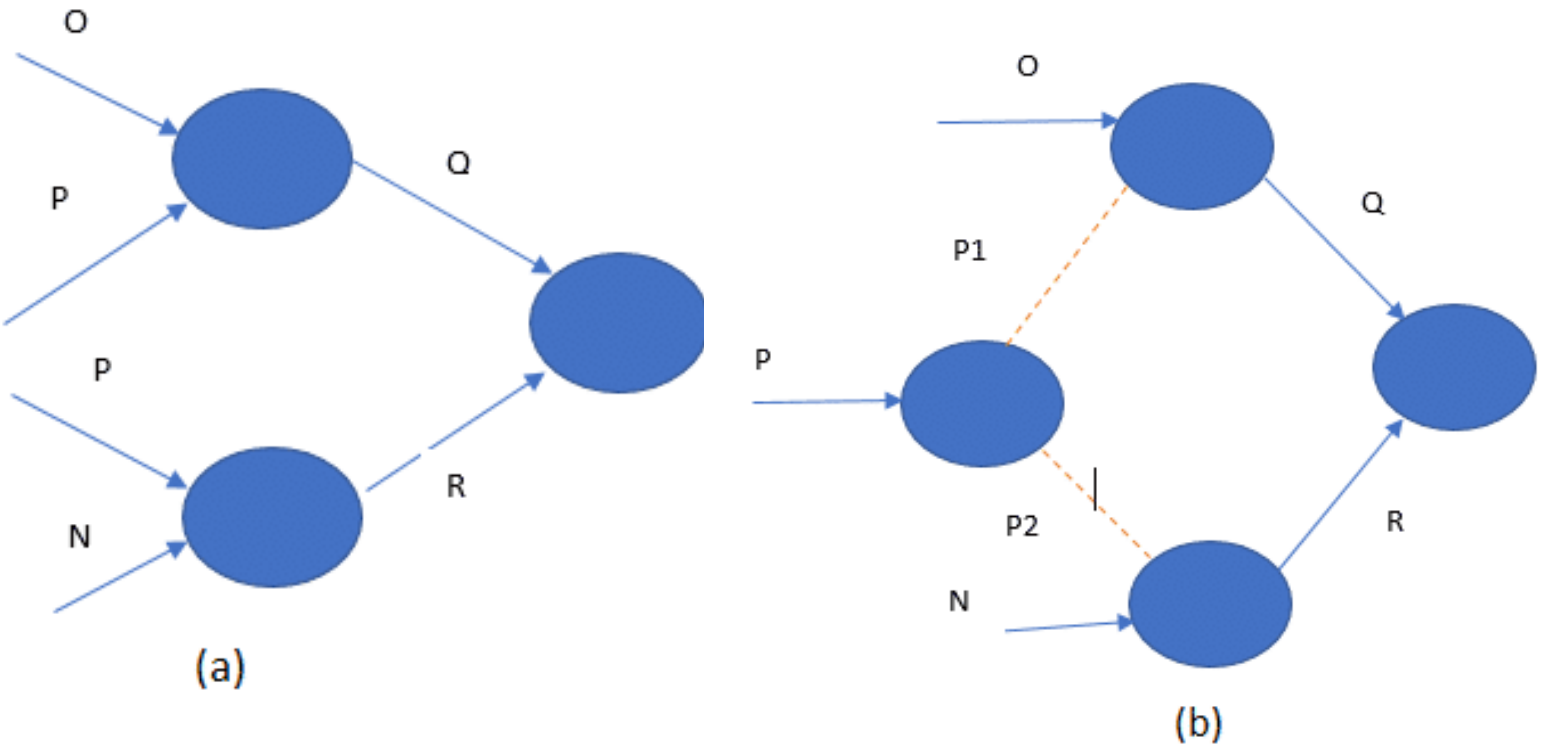


Figure: (a) Illogical Representation of Activities
(b) Logical Representation of Activities using Dummy Activity

NOTE:

- Dummies are used to show predecessor relation but if that relation is already established in the network, then that dummy is redundant and has to be removed.
- If dummy is only incoming/outgoing arrow to/from a node then it can be remove provided there is no logical or grammatical error.

Rules for Using Dummy Activity in Network Analysis

There are several rules for using a dummy activity in network analysis. These rules help to ensure that the network diagram is accurate, easy to understand, and free from errors. The main rules for using a dummy activity in network analysis are:

- Dummy activities are always represented by a dashed line with an arrowhead. They should be labelled with a letter "D" or "dummy".
- Dummy activities always have zero duration and no resources are assigned to them. They are used only to show logical dependencies between activities.

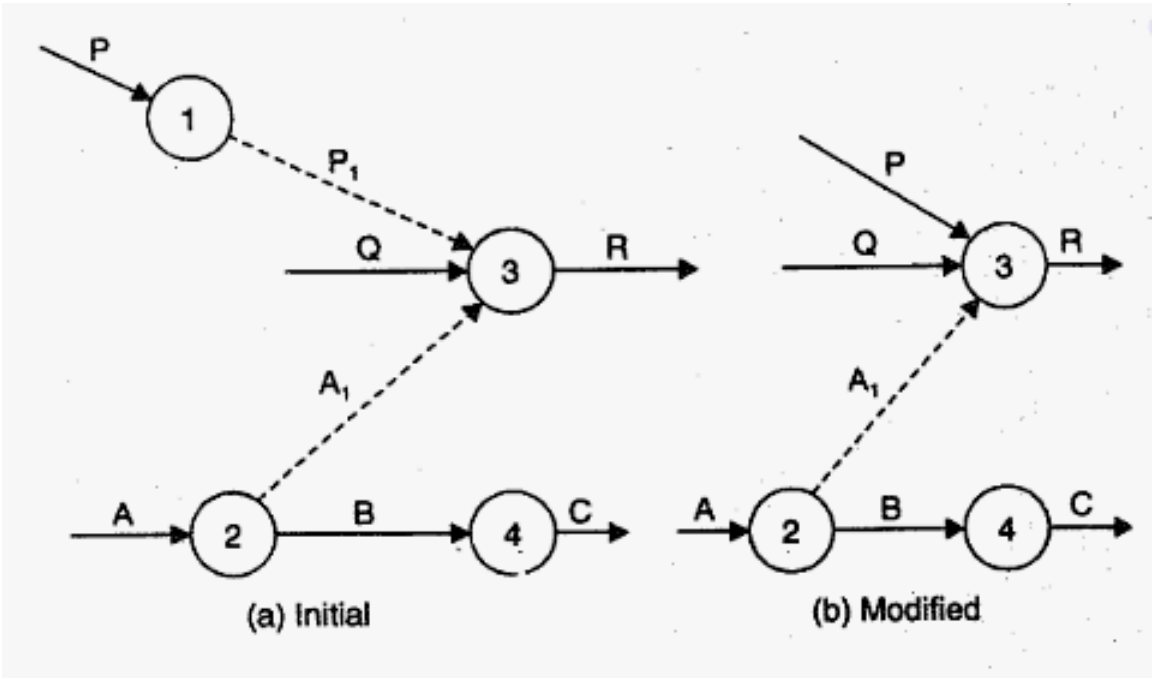
- A dummy activity can only have one predecessor and one successor activity. It cannot have multiple predecessors or successors.
- A dummy activity cannot be the first or last activity in the network diagram. It must connect two other activities and show their dependency.
- A dummy activity cannot have any activity codes or any other attributes that are associated with regular activities.

- Dummy activities should be used sparingly and only when necessary. They should not be used to unnecessarily complicate the network diagram.
- A dummy activity cannot create a loop in the network diagram. It must be used in such a way that it does not create a circular dependency.

By following these rules, project managers can use dummy activities effectively in network analysis to show logical dependencies between activities and ensure that the network diagram accurately reflects the project plan.

While planning a network, a natural question that arises is where to provide dummies. Provision of redundant dummies in the network may create confusion. For that, the simple rule is that during the *initial stage* of developing a network, *liberal use* of dummies should be made to fulfil the requirements of inter-relationships between various activities and between various sets of activities. This may result in the introduction of some unnecessary dummies which can be *removed* by the use of following rules :

1. If a dummy job is the only one emanating from its initial

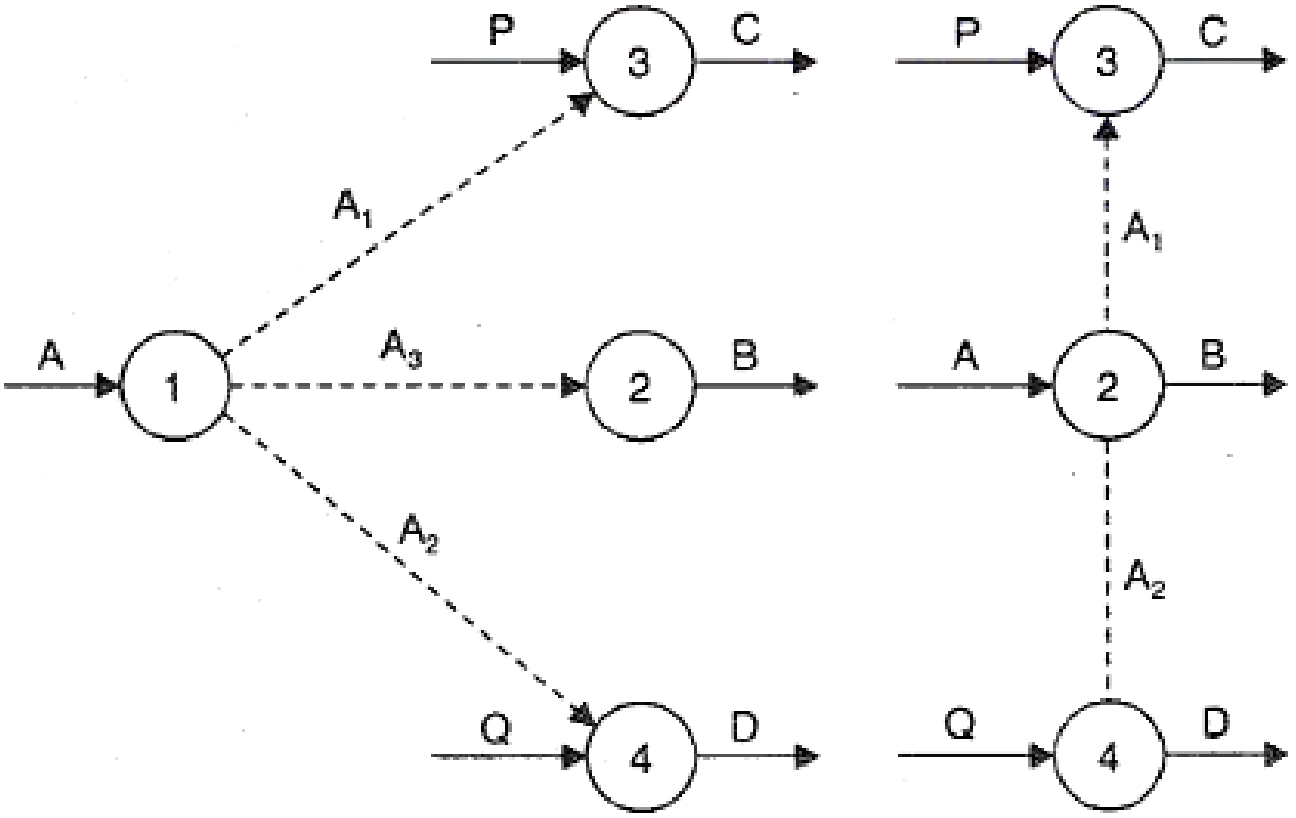


node, it can be removed and the activity terminating at that node can be directly connected to that node to which the dummy was terminating.

For example, consider the initial drawing of the partial network shown in Fig 3.16 (a), the dummy P_1 is the only job emanating from its initial node ; it can therefore be removed and activity P can be directly connected to the forward node (3) as shown in Fig. 3.16 (b). It should be noted that the same treatment cannot be given to dummy A_1, since other jobs or activities (such as B) are also emanating from the same node.

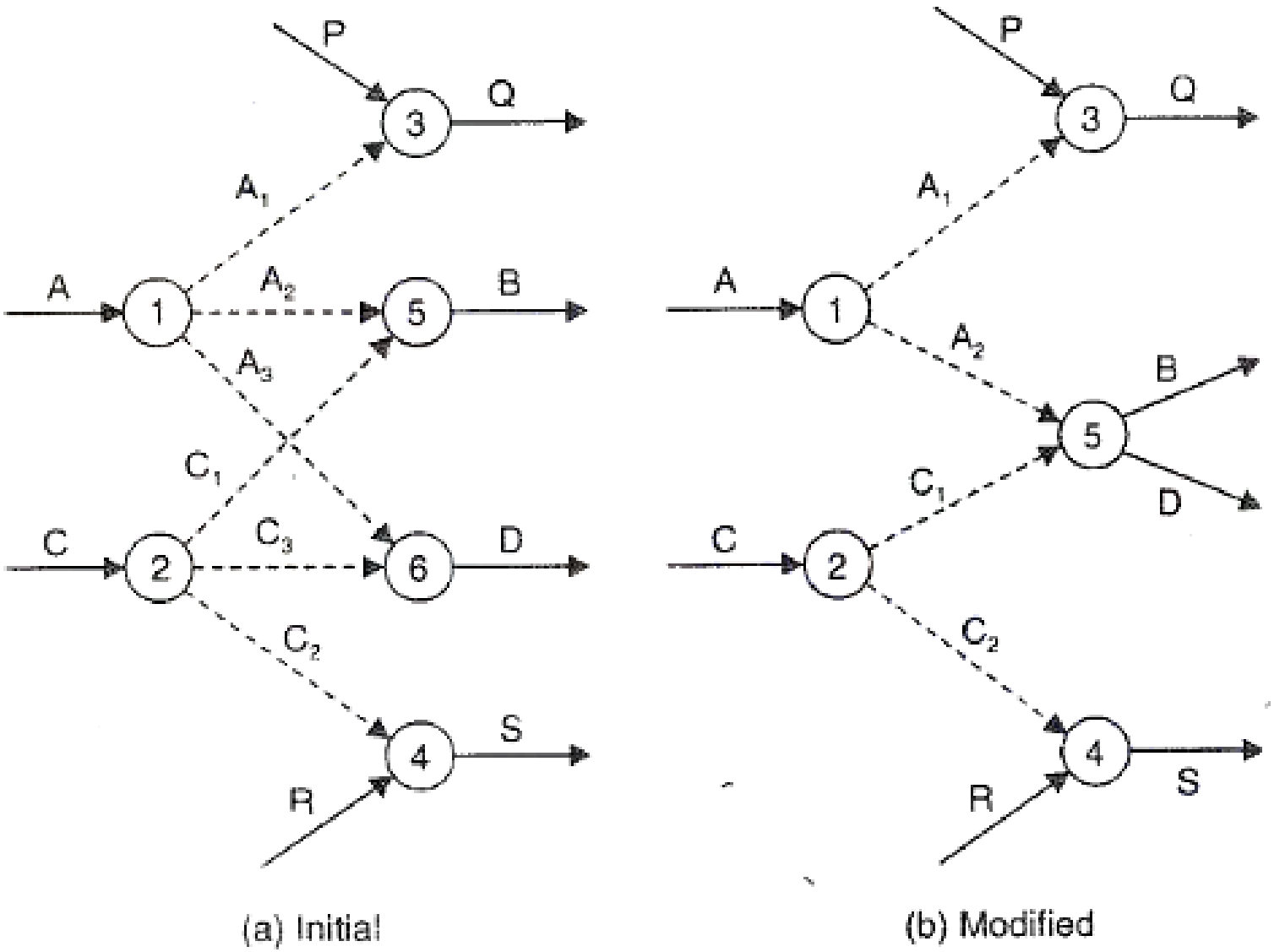
2. If a dummy job is the only one terminating into a node, the dummy can be removed and the two node at the two ends of the dummy can be merged into one.

For example, activity A was initially joined to activities C, D and B by three dummies A_1, A_2 and A_3 respectively. Since dummy A_3 is the only one terminating into node 2, it can be removed, and nodes 1 and 2 situated at the two ends of dummy A_3 can be combined, as shown in Fig. 3.17 (b).



3. If two or more activities, emanating from different nodes, have identical set of predecessors some of which also appear in different predecessor sets of other activities, the two activities should emanate from a single node. This node can then be connected to their predecessor activities by dummies.

For example, consider a partial network situation shown in Fig. 3.18 (*a*), in which two activities B and D emanating from two different nodes 5 and 6 have identical sets of predecessors, some of which also appear in different predecessor sets of other activities. In such case, B and D can be made to emanate from a single node (5) to which the predecessor activities can be joined through dummies. This is shown in Fig. 3.18 (*b*) in which two dummies A_3 and C_3 have been completely eliminated.



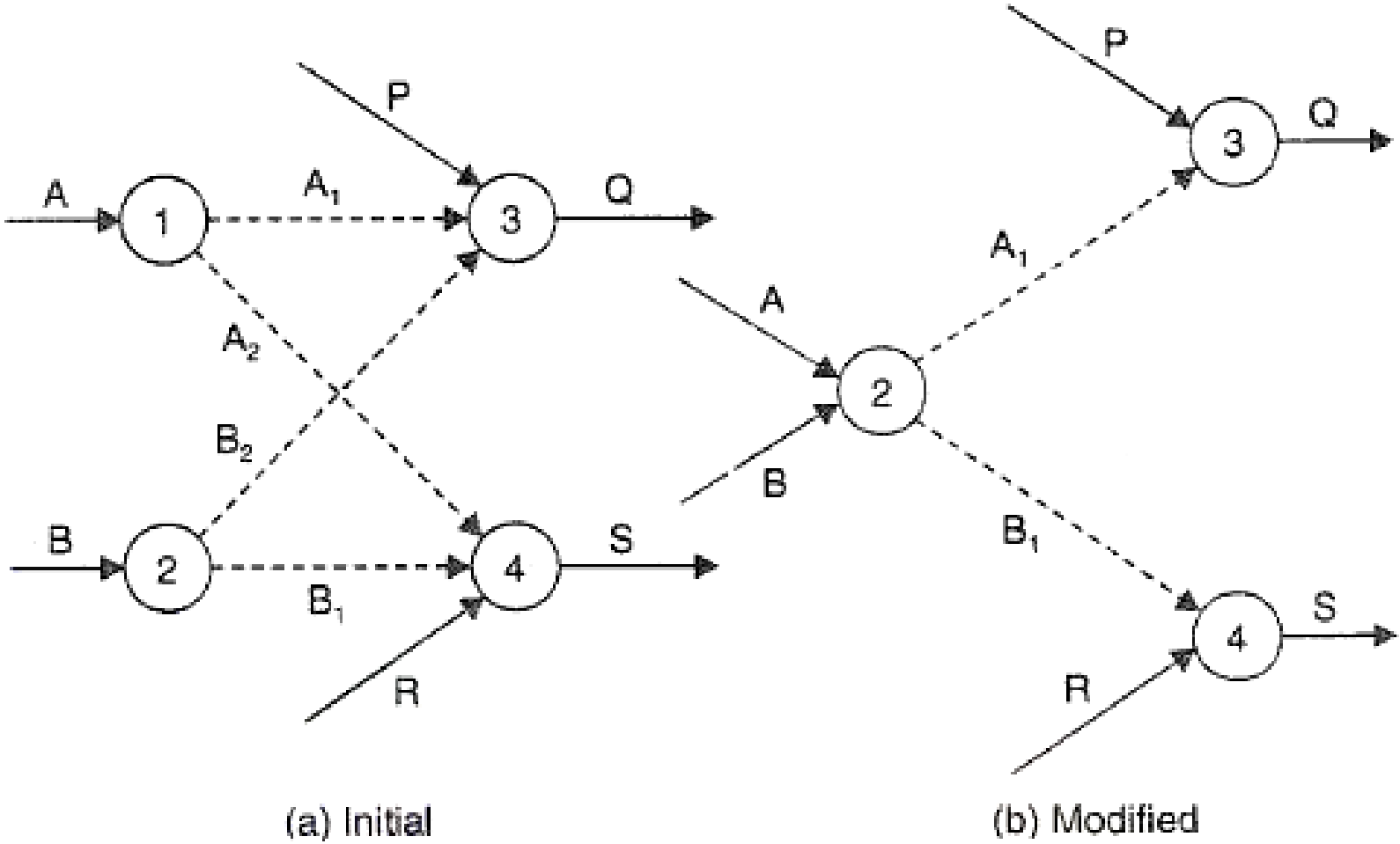
(a) Initial

(b) Modified

4. If two or more activities, terminating into different nodes, have identical set of successors, the latter having other predecessors as well, the two activities should terminate into one single node. This node can then be connected to their successors through appropriate dummies.

For example, consider two activities A and B, terminating into two different nodes 1 and 2, as shown in a partial network situation of Fig. 3.19 (a). Each one of these have identical set of successors Q and S having their other predecessors P and R respectively. In such a case, both A and B can be made to terminate into

one common node (2) and connected to their successors through dummies A_1 and B_1, as shown in Fig. 3.19 (b). Thus two dummies A_2 and B_2 have been eliminated.



5. Such dummies which are used to show predecessor relations already implied by other activities are known as *redundant dummies*, and can be removed.

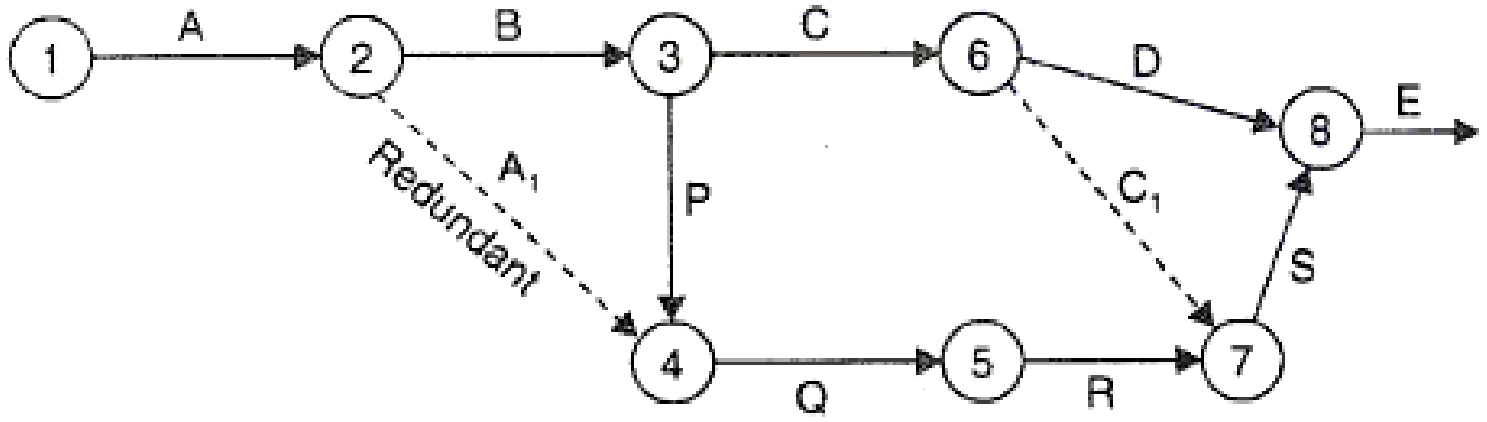
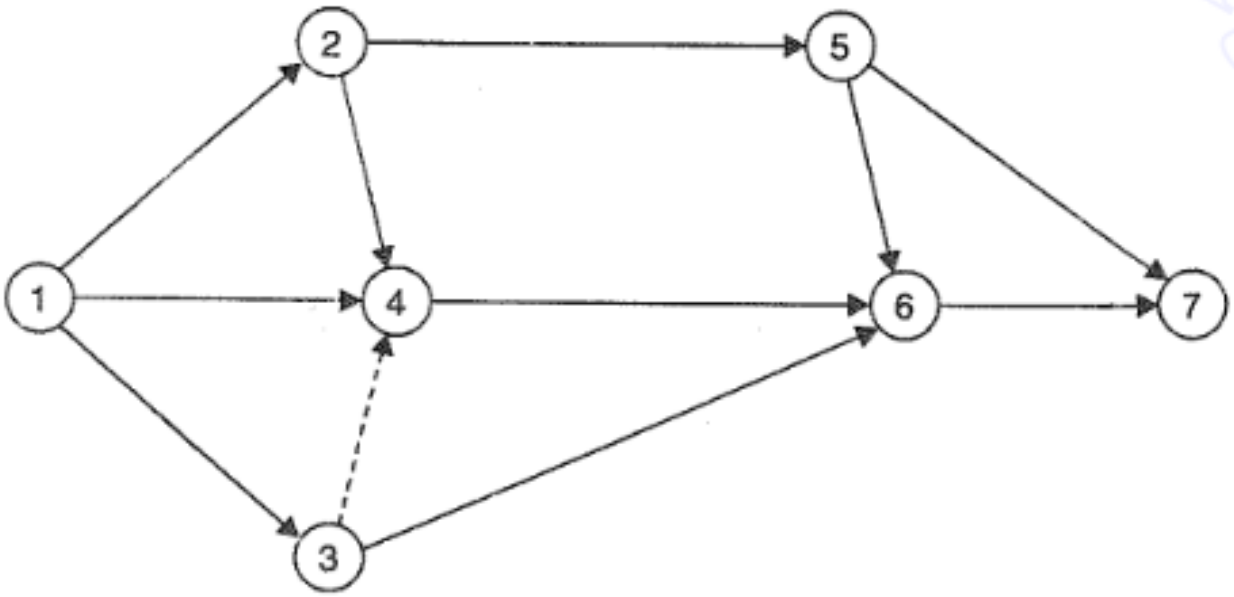


FIG. 3.20

For example, dummy A_1 used to show that activity A is predecessor to activity Q, is redundant since the predecessor relation is already implied by activity P. Hence A_1 can be removed. However, dummy C_1 is not redundant since it has been used to show that activity C is predecessor to activity S; this predecessor relationship was not implied otherwise.

Example: Prepare a table showing immediate predecessor activities and events, predecessor activities and events, immediate successor activities and events, successor activities and events for the various activities and events shown in the figure.



	Immediate predecessor	*Predecessor*	*Immediate Successor*	*Successor*
(a) Activities				
(1–2)	—	—	(2–5)	(2–5), (5–6), (6–7), (5–7)
(1–4)	—	—	(4–6)	(4–6), (6–7)
(1–3)	—	—	(3–6), (3–4)	(3–6), (6–7), (3–4), (4–6)
(2–4)	(1–2)	(1–2)	(4–6)	(4–6), (6–7)
(2–5)	(1–2)	(1–2)	(5–6), (5–7)	(5–6), (5–7), (6–7)
(3–4) Dummy	(1–3)	(1–3)	(4–6)	(4–6), (6–7)
(3–6)	(1–3)	(1–3)	(6–7)	(6–7)
(4–6)	(1–4), (2–4), (3–4)	(1–4), (2–4), (3–4), (1–2), (1–3)	(6–7)	(6–7)
(5–6)	(2–5)	(2–5), (1–2)	(6–7)	(6–7)
(5–7)	(2–5)	(2–5), (1–2)	—	—
(6–7)	(3–6), (4–6), (5–6)	(3–6), (4–6), (5–6), (1–3), (1–4), (1–2), (2–5), (2–4), (3–4)	—	—

(*b*) **Events**				
1	—	—	2, 3, 4	2, 3, 4, 5, 6, 7
2	1	1	4, 5	4, 5, 6, 7
3	1	1	4, 6	4, 6, 7
4	1	1	6	6, 7
5	2	2, 1	6, 7	6, 7
6	5, 4, 3	5, 4, 3, 2, 1	7	7
7	5, 6	5, 6, 2, 4, 3, 1	—	—

Example: To introduce dummy activities in the network shown to identify each activity uniquely

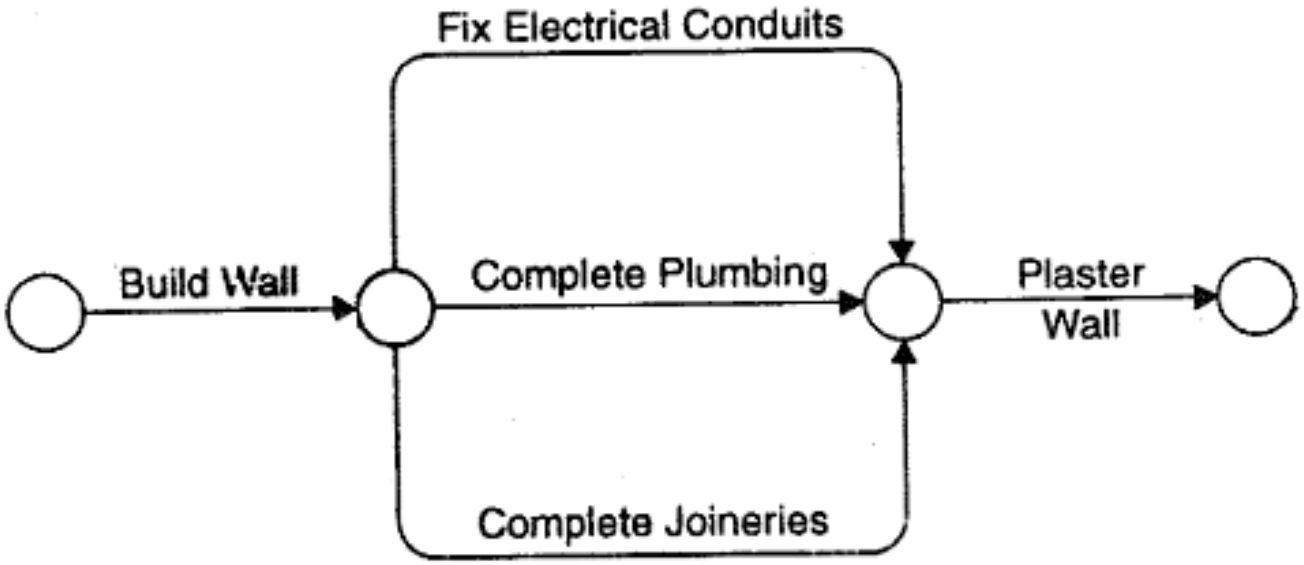
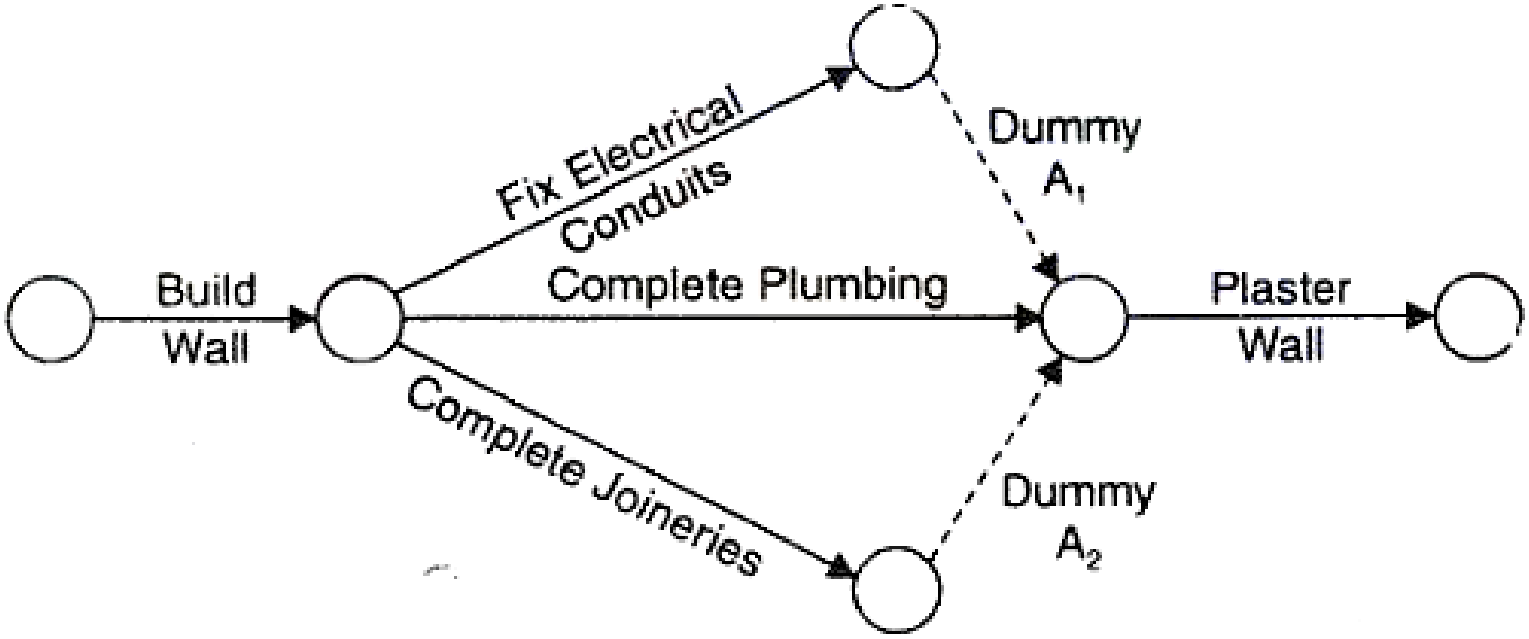


FIG. 3.22

Solution. In the above network, we find that the three activities (fix electrical conduits, complete plumbing and complete joineries) start from one common node and end at some other common node. This gives ambiguous representation, and is grammatically incorrect. Hence, two dummies A_1 and A_2 may be introduced as shown in Fig.

Solution. In the above network, we find that the three activities (fix electrical conduits, complete plumbing and complete joineries) start from one common node and end at some other common node. This gives ambiguous representation, and is grammatically incorrect. Hence, two dummies A_1 and A_2 may be introduced as shown in Fig. 3.23.



Rules for Network

Network diagram rules refer to a set of guidelines and conventions that dictate how to construct a visual representation of a project schedule using nodes (or activities) and arrows (or dependencies) to show the flow of work. Here are some common rules for creating a network diagram:

1. **Activity nodes:** Each activity in the project schedule should be represented by a node (or box) on the network diagram.

2. **Arrows:** Each activity should be connected by arrows to its predecessors (activities that need to be completed before it can start) and successors (activities that can only start after it is completed).
3. **Milestones:** Milestones should be shown as a diamond-shaped node on the network diagram.
4. **Labelling:** Each activity node should have a unique identifier or number, and a label that describes the activity.
5. **Durations:** The duration of each activity should be shown on the network diagram, either within the activity node or as a separate label.

6. **Critical path:** The critical path, which is the longest path through the network diagram, should be clearly marked.
7. **Time scale:** A time scale should be shown along the top or bottom of the network diagram, with tick marks indicating the time intervals.
8. **Legend:** A legend should be provided to explain the symbols used on the network diagram.
9. **Review and update:** The network diagram should be regularly reviewed and updated as the project progresses to ensure it accurately reflects the current state of the project schedule.