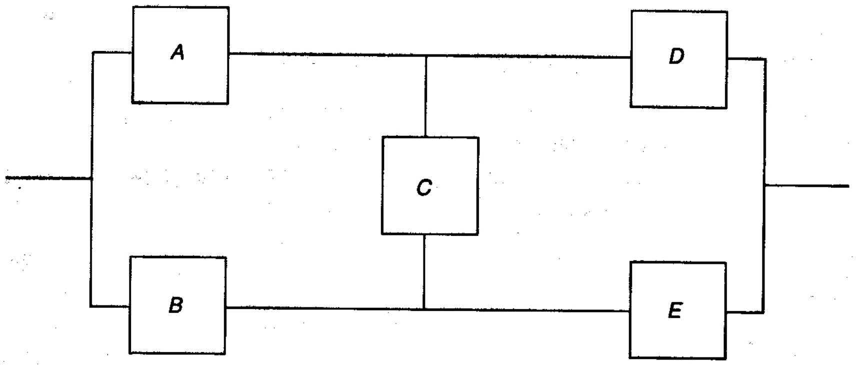
More Complex Configurations

Consider the configuration shown in the figure below. This is neither series nor parallel and not even obtained as a combination of these basic configurations. How can we still compute the reliability? We study three methods, one exact and two approximate.



Exact calculation by exhaustive enumeration

We can list all possible states of the system (see table on the next page) and assign ``up" and ``down" system condition to each state. Then the reliability can be obtained by summing the probability of the ``up" states. This usually yields a long expression, see next page.

Even though the obtained expression may be simplified, this method is only practical for small systems, due to the exponential growth of the number of states. For N components there are 2^N possible states, making exhaustive enumeration a non-scalable solution.

Number of Component Failures	Event	System Condition
0	1. $ABCDE$	Up
1	2. $\overline{A}BCDE$	Up
	3. $A\overline{B}CDE$	Up
	4. $AB\overline{C}DE$	Up
	5. $ABC\overline{D}E$	Up
	6. $ABCD\overline{E}$	Up
2	7. $\overline{AB}CDE$	Down
	8. $\overline{A}B\overline{C}DE$	Up
	9. $\overline{A}BC\overline{D}E$	Up
	10. $\overline{A}BCD\overline{E}$	Up
	11. $A\overline{BC}DE$	Up
	12. $A\overline{B}C\overline{D}E$	Up
	13. $A\overline{B}CD\overline{E}$	Up
	14. $AB\overline{CD}E$	Up
	15. $AB\overline{C}D\overline{E}$	Up
	16. $ABC\overline{DE}$	Down
3	17. $AB\overline{CDE}$	Down
	18. $A\overline{B}C\overline{DE}$	Down
	19. $A\overline{BC}D\overline{E}$	Up
	20. $A\overline{BCD}E$	Down
	21. $\overline{A}BC\overline{DE}$	Down
	22. $\overline{A}B\overline{C}D\overline{E}$	Down
	23. $\overline{A}B\overline{CD}E$	Up
	24. $\overline{AB}CD\overline{E}$	Down
	25. $\overline{AB}C\overline{D}E$	Down
	26. $\overline{ABC}DE$	Down
4	27. $A\overline{BCDE}$	Down
	28. $\overline{A}B\overline{CDE}$	Down
	29. $\overline{AB}C\overline{DE}$	Down
	30. $\overline{ABD}D\overline{E}$	Down
	31. $\overline{ABCD}E$	Down
5	32. $\overline{ABCDE}$	Down

$$
\begin{aligned}
R_{\text{network}} = {} & R_A R_B R_C R_D R_E + (1 - R_A) R_B R_C R_D R_E \\
& + R_A (1 - R_B) R_C R_D R_E + R_A R_B (1 - R_C) R_D R_E \\
& + R_A R_B R_C (1 - R_D) R_E + R_A R_B R_C R_D (1 - R_E) \\
& + (1 - R_A) R_B (1 - R_C) R_D R_E + (1 - R_A) R_B R_C (1 - R_D) R_E \\
& + (1 - R_A) R_B R_C R_D (1 - R_E) + R_A (1 - R_B)(1 - R_C) R_D R_E \\
& + R_A (1 - R_B) R_C (1 - R_D) R_E + R_A (1 - R_B) R_C R_D (1 - R_E) \\
& + R_A R_B (1 - R_C)(1 - R_D) R_E + R_A R_B (1 - R_C) R_D (1 - R_E) \\
& + R_A (1 - R_B)(1 - R_C) R_D (1 - R_E) \\
& + (1 - R_A) R_B (1 - R_C)(1 - R_D) R_E
\end{aligned}
$$

Method of minimal paths

A *minimal path* is a set $\mathcal{P}$ of components with the following properties:

Property 1 If all components in $\mathcal{P}$ are functioning, then the whole system is operational, independently of the status of the other components.

Property 2 $\mathcal{P}$ is minimal with respect to Property 1, that is, no proper subset of $\mathcal{P}$ has Property 1.

Remark: Do not confuse this concept with the well known concept of shortest path. A minimal path in the reliability context does not have to be a shortest path. (In fact, it does not even have to be a path, it can be an arbitrary subset of components with the said properties.)

Example: In the example on page 1 the minimal paths are A-D, B-E, A-C-E and B-C-D. The latter two are not shortest paths.

How can we compute the reliability using minimal paths? Observe that if *any one* of the minimal paths is fully functional, then the system is already operational, by the definition of the minimal path. Therefore, if we can list all minimal paths in the system, then we can regard them as units connected in parallel.

Let us use this approach in our example configuration. For simplicity, let the reliability of each component be denoted by the same letter as the name of the component. Then the reliability of the 4 minimal paths listed above will be AD, BE, ACE, BCD, respectively. Then we can form a parallel configuration from them, thus obtaining:

$$R = 1 - (1 - AD)(1 - BE)(1 - ACE)(1 - BCD).$$

Question: Does this method guarantee an exact result?

Answer: No, it is only an approximation. The reason is that when we compute the reliability of the parallel configuration consisting of the minimal paths, we regard the paths independent. They may not be independent, however, since they can share components.

Question: What is the advantage of this method?

Answer: The number of minimal paths may be much smaller than the number of all states, so the method is more scalable than the exhaustive enumeration of all states.

Method of minimal cuts

This method is similar in spirit to the previous one, but considers cuts rather than paths.

A *minimal cut* is a set $\mathcal{C}$ of components with the following properties:

Property 1 If all components in $\mathcal{C}$ fail, then the whole system fails, independently of the status of the other components.

Property 2 $\mathcal{C}$ is minimal with respect to Property 1, that is, no proper subset of $\mathcal{C}$ has Property 1.

Remark: Observe the duality between the definition of the minimal path and minimal cut. If we replace "functioning/operational" in the definiton of minimal path by "fail", then we obtain the definition of minimal cut.

Example: In our example on page 1 the minimal cuts are A-B, D-E, A-C-E and B-C-D. The latter two are not shortest paths.

To compute the system reliability, observe that if *any one* of the minimal cuts fully fails, then the system must fail, independently of the rest. Therefore, *all* minimal cuts must be operational for the sytem to work. Thus, we can consider the minimal cuts as if they were connected in series.

The probability that a given cut, say A-B, works is $1-(1-A)(1-B)$, since the probability that all components fail in the cut is $(1-A)(1-B)$, so the complement $1-(1-A)(1-B)$ means the cut is operational, i.e., *not all* components fail in it. Since we conceptually consider the cuts as connected in series with each other, therefore, we obtain

$$\begin{aligned} R &= [1-(1-A)(1-B)][1-(1-D)(1-E)] \cdot \\ &\quad \cdot [1-(1-A)(1-C)(1-E)][1-(1-B)(1-C)(1-D)]. \end{aligned}$$

Note: The same comments apply here as for the method of minimal paths: it is only an approximation and its advantage is that it tends to be more scalable than the exhaustive enumeration of all states.

Question: Which method is more advantageous to apply, the minimum paths or minimum cuts?

Answer: It depends on the configuration. If there are many minimum paths but fewer minimum cuts, then the method of minimum cuts is less complex. On the other hand, if there are fewer paths but many cuts, then the method of minimum paths is easier. The previously considered series-parallel and parallel-series configurations show that the number of paths and the number of cuts in a confoiguration can be very different.