



WIKIPEDIA
The Free Encyclopedia

[Main page](#)
[Contents](#)
[Featured content](#)
[Current events](#)
[Random article](#)
[Donate to Wikipedia](#)
[Wikipedia store](#)

Interaction

[Help](#)
[About Wikipedia](#)
[Community portal](#)
[Recent changes](#)
[Contact page](#)

Tools

[What links here](#)
[Related changes](#)
[Upload file](#)
[Special pages](#)
[Permanent link](#)
[Page information](#)
[Wikidata item](#)
[Cite this page](#)

Print/export

[Create a book](#)
[Download as PDF](#)
[Printable version](#)

Languages

[العربية](#)
[Català](#)
[Čeština](#)
[Español](#)
[فارسی](#)
[Français](#)
[Հայերեն](#)
[Italiano](#)
[Polski](#)
[Русский](#)
[Српски / srpski](#)
[ไทย](#)
[Українська](#)
[Tiếng Việt](#)
[中文](#)

[Edit links](#)

[Create account](#) [Log in](#)

Article [Talk](#)

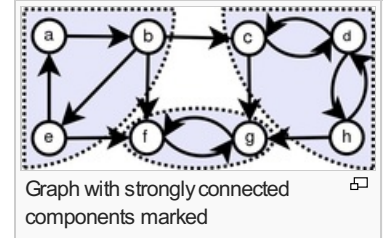
[Read](#) [Edit](#) [View history](#)

Strongly connected component

From Wikipedia, the free encyclopedia

(Redirected from [Strongly connected components](#))

In the mathematical theory of [directed graphs](#), a graph is said to be **strongly connected** if every vertex is [reachable](#) from every other vertex. The **strongly connected components** of an arbitrary directed graph form a [partition](#) into subgraphs that are themselves strongly connected. It is possible to test the strong connectivity of a graph, or to find its strongly connected components, in [linear time](#).



Contents [\[hide\]](#)

- [1 Definitions](#)
- [2 Algorithms](#)
- [3 Applications](#)
- [4 Related results](#)
- [5 See also](#)
- [6 References](#)
- [7 External links](#)

Definitions [\[edit\]](#)

A [directed graph](#) is called **strongly connected** if there is a [path](#) in each direction between each pair of vertices of the graph. In a directed graph *G* that may not itself be strongly connected, a pair of vertices *u* and *v* are said to be strongly connected to each other if there is a path in each direction between them.

The [binary relation](#) of being strongly connected is an [equivalence relation](#), and the [induced subgraphs](#) of its [equivalence classes](#) are called **strongly connected components**. Equivalently, a **strongly connected component** of a directed graph *G* is a subgraph that is strongly connected, and is [maximal](#) with this property: no additional edges or vertices from *G* can be included in the subgraph without breaking its property of being strongly connected. The collection of strongly connected components forms a [partition](#) of the set of vertices of *G*.

If each strongly connected component is [contracted](#) to a single vertex, the resulting graph is a [directed acyclic graph](#), the **condensation** of *G*. A directed graph is acyclic if and only if it has no strongly connected subgraphs with more than one vertex, because a directed cycle is strongly connected and every nontrivial strongly connected component contains at least one directed cycle.

Algorithms [\[edit\]](#)

Several algorithms can compute strongly connected components in [linear time](#).

- [Kosaraju's algorithm](#) uses two passes of [depth first search](#). The first, in the original graph, is used to choose the order in which the outer loop of the second depth first search tests vertices for having been visited already and recursively explores them if not. The second depth first search is on the [transpose graph](#) of the original graph, and each recursive exploration finds a single new strongly connected component.^[1] It is named after [S. Rao Kosaraju](#), who described it (but did not publish his results) in 1978; [Micha Sharir](#) later published it in 1981.^[2]
- [Tarjan's strongly connected components algorithm](#), published by [Robert Tarjan](#) in 1972,^[3] performs a single pass of depth first search. It maintains a [stack](#) of vertices that have been explored by the search but not yet assigned to a component, and calculates "low numbers" of each vertex (an index number of the highest ancestor reachable in one step from a descendant of the vertex) which it uses to determine when a set of vertices should be popped off the stack into a new component.
- The [path-based strong component algorithm](#) uses a depth first search, like Tarjan's algorithm, but with two stacks. One of the stacks is used to keep track of the vertices not yet assigned to components, while the other keeps track of the current path in the depth first search tree. The first linear time version of this algorithm was published by [Edsger W. Dijkstra](#) in 1976.^[4]

Although Kosaraju's algorithm is conceptually simple, Tarjan's and the path-based algorithm are favoured in practice since they require only one [depth-first search](#) rather than two.

Applications [\[edit\]](#)

Algorithms for finding strongly connected components may be used to solve [2-satisfiability](#) problems (systems of Boolean variables with constraints on the values of pairs of variables): as [Aspvall, Plass & Tarjan \(1979\)](#) showed, a [2-satisfiability](#) instance is unsatisfiable if and only if there is a variable *v* such that *v* and its complement are both contained in the same strongly connected component of the [implication graph](#) of the instance.^[5]

Strongly connected components are also used to compute the [Dulmage–Mendelsohn decomposition](#), a classification of the edges of a [bipartite graph](#), according to whether or not they can be part of a [perfect matching](#) in the graph.^[6]

Related results [\[edit\]](#)

A directed graph is strongly connected if and only if it has an [ear decomposition](#), a partition of the edges into a sequence of directed paths and cycles such that the first subgraph in the sequence is a cycle, and each subsequent subgraph is either a cycle sharing one vertex with previous subgraphs, or a path sharing its two endpoints with previous subgraphs.

According to [Robbins' theorem](#), an undirected graph may be [oriented](#) in such a way that it becomes strongly connected, if and only if it is [2-edge-connected](#). One way to prove this result is to find an ear decomposition of the underlying undirected graph and then orient each ear consistently.^[7]

See also [\[edit\]](#)

- [Connected component](#)
- [Modular decomposition](#)

References [\[edit\]](#)

- ↑ [Thomas H. Cormen](#), [Charles E. Leiserson](#), [Ronald L. Rivest](#), and [Clifford Stein](#). *Introduction to Algorithms*, Second Edition. MIT Press and McGraw-Hill, 2001. ISBN 0-262-03293-7. Section 22.5, pp. 552–557.
- ↑ [Micha Sharir](#). A strong connectivity algorithm and its applications to data flow analysis. *Computers and Mathematics with Applications* 7(1):67–72, 1981.
- ↑ [Tarjan, R. E.](#) (1972), "Depth-first search and linear graph algorithms", *SIAM Journal on Computing* **1** (2): 146–160, doi:10.1137/0201010
- ↑ [Dijkstra, Edsger](#) (1976), *A Discipline of Programming*, NJ: Prentice Hall, Ch. 25.
- ↑ [Aspvall, Bengt](#); [Plass, Michael F.](#); [Tarjan, Robert E.](#) (1979), "A linear-time algorithm for testing the truth of certain quantified boolean formulas", *Information Processing Letters* **8** (3): 121–123, doi:10.1016/0020-0190(79)90002-4 .
- ↑ [Dulmage, A. L.](#) & [Mendelsohn, N. S.](#) (1958), "Coverings of bipartite graphs", *Canad. J. Math.* **10**: 517–534, doi:10.4153/cjm-1958-052-0 .
- ↑ [Robbins, H. E.](#) (1939), "A theorem on graphs, with an application to a problem on traffic control", *American Mathematical Monthly* **46**: 281–283, JSTOR 2303897 .

External links [\[edit\]](#)

- [Java implementation for computation of strongly connected components](#) in the jBPT library (see `StronglyConnectedComponents` class).
- [C implementation of Strongly Connected Components](#)

Categories: [Graph connectivity](#) | [Graph algorithms](#) | [Directed graphs](#)

This page was last modified on 26 December 2014, at 05:36.

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. By using this site, you agree to the [Terms of Use](#) and [Privacy Policy](#). Wikipedia® is a registered trademark of the [Wikimedia Foundation, Inc.](#), a non-profit organization.

[Privacy policy](#) [About Wikipedia](#) [Disclaimers](#) [Contact Wikipedia](#) [Developers](#) [Mobile view](#)

