Crash Course on Graph Theory

A *graph* is a pair (V, E), where V is a set of vertices, and $E = \{(v_1, v_2) \mid v_1, v_2 \in V\}$ is a set of edges between vertices in V.

A *simple graph* is a graph (V, E) in which every pair of vertices $v_1, v_2 \in V$ is connected by at most one edge $(v_1, v_2) \in E$.

A *path* $(v_1, v_2, ..., v_n)$ from vertex $v_1 \in V$ to vertex $v_n \in V$ in a graph (V, E) is a set of edges $(v_1, v_2), (v_2, v_3), ..., (v_{n-1}, v_n) \in E$.

A *Hamiltonian path* in a graph (V, E) is a path that visits every vertex in V only once.

A *semi-Hamiltonian graph* is a graph with a Hamiltonian path.

A *directed graph* is a graph in which each edge is unidirectional from one vertex to the next.

A *directed acyclic graph* is a directed graph in which no path starts and ends at the same vertex.

A *source* in a directed graph is a vertex without incoming edges.

A *sink* in a directed graph is a vertex without outgoing edges.

An *st-digraph* is a directed acyclic graph with only one source and only one sink.

Mathematical Synopsis: Definitions and Proofs

Definition 1: An *SIT code* $\overline{X}$ of a string X is a string $t_1t_2...t_m$ such that $X = D(t_1)...D(t_m)$, where the decoding function $D : t \rightarrow D(t)$ takes one of the following forms:

I-form:	$n * (\overline{y})$	$\rightarrow$	$yyy...y$	(n times y; $n \geq 2$)
S-form:	$S[\overline{(\overline{x_1})(\overline{x_2})...(\overline{x_n})}, (\overline{p})]$	$\rightarrow$	$x_1x_2...x_n\ p\ x_n...x_2x_1$	($n \geq 1$)
A-form:	$\langle(\overline{y})\rangle/\langle\overline{(\overline{x_1})(\overline{x_2})...(\overline{x_n})}\rangle$	$\rightarrow$	$yx_1\ yx_2\ ...\ yx_n$	($n \geq 2$)
A-form:	$\langle\overline{(\overline{x_1})(\overline{x_2})...(\overline{x_n})}\rangle/\langle(\overline{y})\rangle$	$\rightarrow$	$x_1y\ x_2y\ ...\ x_ny$	($n \geq 2$)
Otherwise:	$D(t) = t$			

for strings y, p, and x_i ($i = 1, 2, ..., n$). The code parts $(\overline{y})$, $(\overline{p})$, and $(\overline{x_i})$ are called *chunks*; the chunk $(\overline{y})$ in an I-form or an A-form is called a *repeat*; the chunk $(\overline{p})$ in an S-form is called a *pivot*, which as a limit case may be empty; the chunk string $(\overline{x_1})(\overline{x_2})...(\overline{x_n})$ in an S-form is called an *S-argument* consisting of *S-chunks* $(\overline{x_i})$; and the chunk string $(\overline{x_1})(\overline{x_2})...(\overline{x_n})$ in an A-form is called an *A-argument* consisting of *A-chunks* $(\overline{x_i})$.

Definition 2: A *hyperstring* is a simple semi-Hamiltonian directed acyclic graph (V, E) with a labeling of the edges in E such that, for all vertices $i, j, p, q \in V$:

$$\text{either } \pi(i, j) = \pi(p, q) \text{ or } \pi(i, j) \cap \pi(p, q) = \emptyset,$$

where a *substring set* $\pi(v_1, v_2)$ is the set of label strings represented by the paths $(v_1, ..., v_2)$ in an edge-labeled directed acyclic graph. In a hyperstring, the subgraph formed by the vertices and edges in these paths $(v_1, ..., v_2)$ is called a *hypersubstring*.

Definition 3: For a string $T = s_1s_2...s_N$, the *A-graph* $\mathcal{A}(T)$ is a simple directed acyclic graph (V, E) with $V = \{1, 2, .., N+1\}$ and, for all $1 \leq i < j \leq N$, edges (i, j) and $(j, N+1)$ labeled with, respectively, the chunks $(s_i...s_{j-1})$ and $(s_j...s_N)$ if and only if $s_i = s_j$.

Definition 4: A *diafix* of a string $T = s_1s_2...s_N$ is a substring $s_{i+1}...s_{N-i}$ $(0 \leq i < N/2)$.

Definition 5: For a string $T = s_1s_2...s_N$, the *S-graph* $\mathcal{S}(T)$ is a simple directed acyclic graph (V, E) with $V = \{1, 2, .., \lfloor N/2\rfloor + 2\}$ and, for all $1 \leq i < j < \lfloor N/2\rfloor + 2$, edges (i, j) and $(j, \lfloor N/2\rfloor + 2)$ labeled with, respectively, the chunk $(s_i...s_{j-1})$ and the possibly empty chunk $(s_j...s_{N-j+1})$ if and only if $s_i...s_{j-1} = s_{N-j+2}...s_{N-i+1}$.

Theorem 1. *The A-graph $\mathcal{A}(T)$ for a string $T = s_1 s_2 ... s_N$ consists of at most $N+1$ disconnected vertices and at most $\lfloor N/2 \rfloor$* independent *subgraphs (i.e., subgraphs that share only the sink vertex $N+1$) each of which is a hyperstring.*

Proof: First, by *Definition 3*, vertex i ($i \leq N$) in $\mathcal{A}(T)$ does not have incoming or outgoing edges if and only if s_i is a unique element in T. Since T contains at most N unique elements, $\mathcal{A}(T)$ contains at most $N+1$ disconnected vertices, as required.

Second, let $s_{i_1}, s_{i_2}, ..., s_{i_n}$ ($i_p < i_{p+1}$) be a complete set of identical elements in T. Then, by *Definition 3*, the vertices $i_1, i_2, ..., i_n$ in $\mathcal{A}(T)$ are connected with each other and with vertex $N+1$ but not with any other vertex. Hence, the subgraph on the vertices $i_1, i_2, ..., i_n, N+1$ forms an independent subgraph. For every complete set of identical elements in T, n may be as small as 2, so that $\mathcal{A}(T)$ contains at most $\lfloor N/2 \rfloor$ independent subgraphs, as required.

Third, to be hyperstrings, the independent subgraphs must at least be semi-Hamiltonian. Now, let $s_{i_1}, s_{i_2}, ..., s_{i_n}$ ($i_p < i_{p+1}$) again be a complete set of identical elements in T. Then, by *Definition 3*, $\mathcal{A}(T)$ contains edges (i_p, i_{p+1}), $p = 1, 2, ..., n-1$, and it contains edge $(i_n, N+1)$. Together, these edges form a Hamiltonian path through the independent subgraph on the vertices $i_1, i_2, ..., i_n, N+1$, as required.

Fourth, the only thing left to prove is that the substring sets are pairwise either identical or disjunct (see *Definition 2*). Now, for $i < j$ and $k \geq 1$, let substring sets $\pi(i, i+k)$ and $\pi(j, j+k)$ in $\mathcal{A}(T)$ be not disjunct, that is, let them share at least one chunk string. Then, the substrings $s_i ... s_{i+k-1}$ and $s_j ... s_{j+k-1}$ of T are necessarily identical and, also necessarily, $s_i = s_{i+k}$ and either $s_j = s_{j+k}$ or $j+k = N+1$. Hence, by *Definition 3*, these identical substrings of T yield, in $\mathcal{A}(T)$, edges $(i, i+k)$ and $(j, j+k)$ labeled with the identical chunks $(s_i ... s_{i+k-1})$ and $(s_j ... s_{j+k-1})$, respectively. Furthermore, obviously, these identical substrings of T can be chunked into exactly the same strings of two or more identically beginning chunks. By *Definition 3*, all these chunks are represented in $\mathcal{A}(T)$, so that each of these chunkings is represented not only by a path $(i, ..., i+k)$ but also by a path $(j, ..., j+k)$. This implies that the substring sets $\pi(i, i+k)$ and $\pi(j, j+k)$ are identical. The foregoing holds not only for the entire A-graph but, because of their independence, also for every independent subgraph. Hence, in sum, every independent subgraph is a hyperstring as required.

Lemma 1 (Used in *Theorem 2*). *In the S-graph $\mathcal{S}(T)$ for a string $T = s_1 s_2 ... s_N$, the substring sets $\pi(v_1, v_2)$ ($1 \leq v_1 < v_2 < \lfloor N/2 \rfloor + 2$) are pairwise either identical or disjunct.*

Proof: Let, for $i < j$ and $k \geq 1$, substring sets $\pi(i, i{+}k)$ and $\pi(j, j{+}k)$ in $\mathcal{S}(T)$ be nondisjunct, that is, let them share at least one S-chunk string. Then, the substrings $s_i ... s_{i+k-1}$ and $s_j ... s_{j+k-1}$ in the left-hand half of T are necessarily identical to each other. Furthermore, by *Definition 5*, the substring in each chunk of these S-chunk strings is identical to its symmetrically positioned counterpart in the right-hand half of T, so that also the substrings $s_{N-i-k+2} ... s_{N-i+1}$ and $s_{N-j-k+2} ... s_{N-j+1}$ in the right-hand half of T are identical to each other. Hence, the diafixes $D_1 = s_i ... s_{N-i+1}$ and $D_2 = s_j ... s_{N-j+1}$ can be written as

$$D_1 = s_i ... s_{i+k-1} \;\; p_1 \;\; s_{N-i-k+2} ... s_{N-i+1}$$
$$D_2 = s_i ... s_{i+k-1} \;\; p_2 \;\; s_{N-i-k+2} ... s_{N-i+1}$$

with $p_1 = s_{i+k} ... s_{N-i-k+1}$ and $p_2 = s_{j+k} ... s_{N-j-k+1}$. Now, by means of any S-chunk string C in $\pi(i, i + k)$, diafix D_1 can be encoded into the covering S-form $S[C, (p_1)]$. If, in this S-form, the pivot (p_1) is replaced by (p_2), then one gets the covering S-form $S[C, (p_2)]$ for diafix D_2. This implies that any S-chunk string in $\pi(i, i + k)$ is in $\pi(j, j + k)$, and vice versa. Hence, nondisjunct substring sets $\pi(i, i + k)$ and $\pi(j, j + k)$ are identical as required.

Lemma 2 (Used in *Lemma 3*). *Let the strings* $c_1 = s_1s_2...s_k$ *and* $c_2 = s_1s_2...s_p$ $(k < p)$ *be such that* c_2 *can be written in the following two ways:*

$$c_2 = c_1X \text{ with } X = s_{k+1}...s_p$$
$$c_2 = Yc_1 \text{ with } Y = s_1...s_{p-k}$$

Then, $X = Y$ *if* $q = p/(p-k)$ *is an integer; otherwise* $Y = VW$ *and* $X = WV$*, where* $V = s_1...s_r$ *and* $W = s_{r+1}...s_{p-k}$*, with* $r = p - \lfloor q \rfloor (p-k)$.

Proof: Take q, r, V, and W as given above, and distinguish between the next three cases.
(1) If $1 < q < 2$, then $c_2 = c_1Wc_1$, so that $Y = c_1W$ and $X = Wc_1$. Then, too, $r = k$, so that $c_1 = V$. Hence, $Y = VW$ and $X = WV$, as required in this case (q is noninteger).
(2) If $q = 2$, then $c_2 = c_1c_1$. Hence, $X = Y = c_1$, as required in this case (q is integer).
(3) If $q > 2$, then the two copies of c_1 in c_2 overlap each other as follows:

$$\begin{array}{ccccccccccccc} c_2 & = & c_1X & = & s_1 & \dots & s_{p-k} & s_{p-k+1} & \dots & s_k & s_{k+1} & \dots & s_p \\ c_2 & = & Yc_1 & = & & Y & & s_1 & \dots & s_{2k-p} & s_{2k-p+1} & \dots & s_k \end{array}$$

Hence, $s_i = s_{p-k+i}$ for $i = 1, 2, ..., k$. That is, c_2 is a prefix of an infinite repetition of Y. Now, distinguish between integer q and noninteger q as follows.
(3a) If q is an integer, then c_2 is a q-fold repetition of Y, that is, $c_2 = YY...Y$. This implies (because also $c_2 = Yc_1$) that c_1 is a $(q-1)$-fold repetition of Y, so that c_2 can also be written as $c_2 = c_1Y$. This implies that $X = Y$, as required.
(3b) If q is not an integer, then c_2 is a $\lfloor q \rfloor$-fold repetition of Y plus a residual prefix V of Y, that is, $c_2 = YY...YV$. Now, $Y = VW$, so that c_2 can also be written as $c_2 = VWVW...VWV$. This implies (because also $c_2 = Yc_1 = VWc_1$) that $c_1 = VW...VWV$, that is, c_1 is a $(\lfloor q \rfloor - 1)$-fold repetition of $Y = VW$ plus a residual part V. This, in turn, implies that c_2 can also be written as $c_2 = c_1WV$, so that $X = WV$, as required.

Lemma 3 (Used in *Theorem 2*). *Let $\mathcal{S}(T)$ be the S-graph for a string $T = s_1 s_2 ... s_N$. Then:*
(A) *If $\mathcal{S}(T)$ contains edges $(i, i+k)$ and $(i, i+p)$, with $k < p < \lfloor N/2 \rfloor + 2 - i$, then it also contains a path $(i+k, ..., i+p)$.*
(B) *If $\mathcal{S}(T)$ contains edges $(i-k, i)$ and $(i-p, i)$, with $k < p$ and $i < \lfloor N/2 \rfloor + 2$, then it also contains a path $(i-p, ..., i-k)$.*

Proof: **(A)** Edge $(i, i+k)$ represents the S-chunk $(c_1) = (s_i ... s_{i+k-1})$, and edge $(i, i+p)$ represents the S-chunk $(c_2) = (s_i ... s_{i+p-1})$. This implies that the diafix $D = s_i ... s_{N-i+1}$ of T can be written in the following two ways:

$$\begin{array}{ccccc} D & = & c_2 & \dots & c_2 \\ D & = & c_1 & \dots & c_1 \end{array}$$

This implies that c_2 (which is longer than c_1) can be written in the following two ways:

$$c_2 = c_1 X \text{ with } X = s_{i+k} ... s_{i+p-1}$$
$$c_2 = Y c_1 \text{ with } Y = s_i ... s_{i+p-k-1}$$

Hence, by *Lemma 2*, either $X = Y$ or $Y = VW$ and $X = WV$ for some V and W. If $X = Y$, then $D = c_1 Y ... Y c_1$ so that, by *Definition 5*, Y is an S-chunk represented by an edge that yields a path $(i+k, ..., i+p)$ as required. If $Y = VW$ and $X = WV$, then $D = c_1 WV ... VW c_1$ so that, by *Definition 5*, W and V are S-chunks represented by subsequent edges that yield a path $(i+k, ..., i+p)$ as required.
(B) This time, edge $(i-k, i)$ represents the S-chunk $(c_1) = (s_{i-k} ... s_{i-1})$, and edge $(i-p, i)$ represents the S-chunk $(c_2) = (s_{i-p} ... s_{i-1})$. This implies that the diafix $D = s_{i-p} ... s_{N-i+p+1}$ of T can be written in the following two ways:

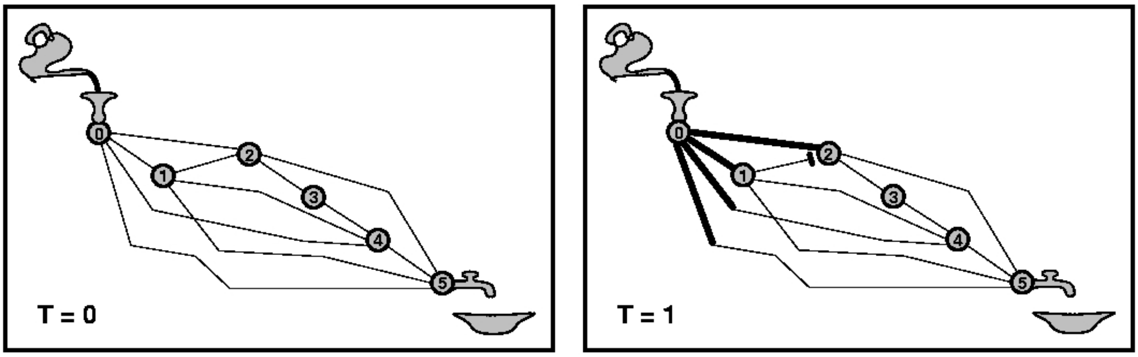
$$\begin{array}{ccccc} D & = & c_2 & \dots & c_2 \\ D & = & Y c_1 & \dots & c_1 X \end{array}$$

with $X{=}s_{i-p+k} ... s_{i-1}$ and $Y{=}s_{i-p} ... s_{i-k-1}$. Hence, as before, $c_2 = c_1 X$ and $c_2 = Y c_1$, so that, by *Lemma 2*, either $X = Y$ or $Y = VW$ and $X = WV$ for some V and W. This implies either $D = Y c_1 ... c_1 Y$ or $D = VW c_1 ... c_1 WV$. Hence, this time, *Definition 5* implies that both cases yield a path $(i-p, ..., i-k)$ as required.

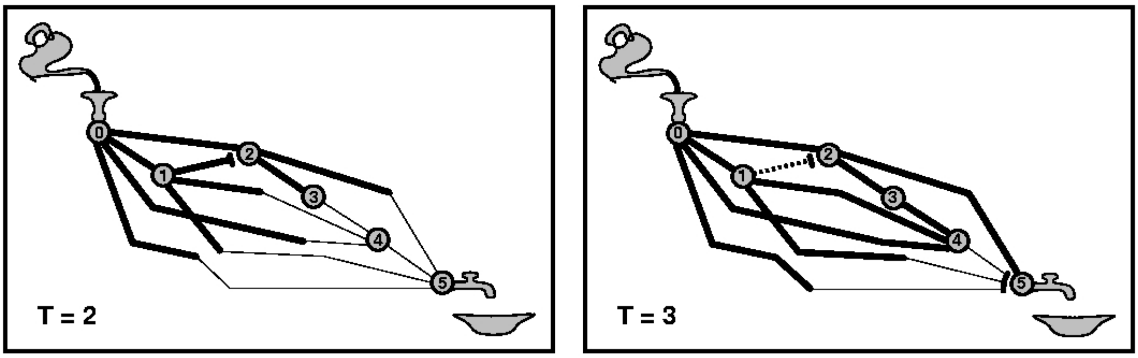
Theorem 2. *The S-graph $\mathcal{S}(T)$ for a string $T = s_1 s_2 ... s_N$ consists of at most $\lfloor N/2 \rfloor + 2$ disconnected vertices and at most $\lfloor N/4 \rfloor$ independent subgraphs that, without the sink vertex $\lfloor N/2 \rfloor + 2$ and its incoming pivot edges, form one disconnected hyperstring each.*

Proof: From *Definition 5*, it is obvious that there may be disconnected vertices and that their number is at most $\lfloor N/2 \rfloor + 2$, so let us turn to the more interesting part. If $\mathcal{S}(T)$ contains one or more paths $(i, ..., j)$ $(i < j < \lfloor N/2 \rfloor + 2)$ then, by *Lemma 3*, one of these paths visits every vertex v with $i < v < j$ and v connected to i or j. This implies that, without the pivot edges and apart from disconnected vertices, $\mathcal{S}(T)$ consists of disconnected semi-Hamiltonian subgraphs. Obviously, the number of such subgraphs is at most $\lfloor N/4 \rfloor$, and if these subgraphs are expanded to include the pivot edges, they form one independent subgraph each. More importantly, by *Lemma 1*, these disconnected semi-Hamiltonian subgraphs form one hyperstring each, as required.

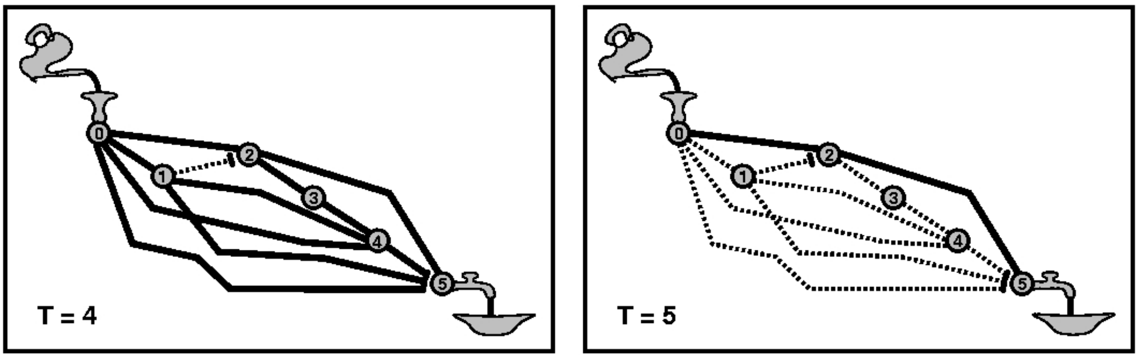
Parallel Distributed Processing Implementation of Dijkstra's Shortest-Path Method



At time $T = 0$, a fluid starts to be poured into node 0. The fluid is such that it hardens within 1 time unit after it stops flowing. Every link between two nodes is a soft tube that expands as the fluid enters and that consists of straight segments having slopes such that the fluid takes 1 time unit to cross one segment. Every node has a separate outlet for each outgoing tube but only one inlet for all incoming tubes. An inlet has about the same cross section as one fluid-filled tube. Hence, when the fluid reaches an inlet through one or more tubes, the remaining tubes are automatically sealed off. Thus, at time $T = 1$, the fluid reaches node 2, sealing off the tube between nodes 1 and 2.



At time $T = 2$, the fluid has filled this dead-end tube between nodes 1 and 2, and the then nonflowing fluid therein has hardened at time $T = 3$. By then, the fluid has also already reached node 5.



Around time $T = 4$, there is still some filling of dead-end tubes and hardening of the fluid therein, but as of time $T = 5$, the only remaining flow is through the shortest path between nodes 0 and 5. Thus, in $O(N)$ time units, a shortest path is selected from among $O(2^N)$ possible paths.