SAND2018-9954PE

Notional Reliabilities of Systems

Michael Richard Livesay [1]

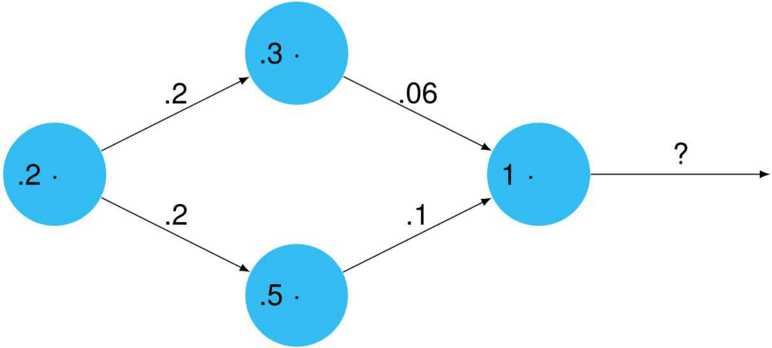
September 6, 2018

[1] Sandia National Laboratory, Albuquerque, NM.

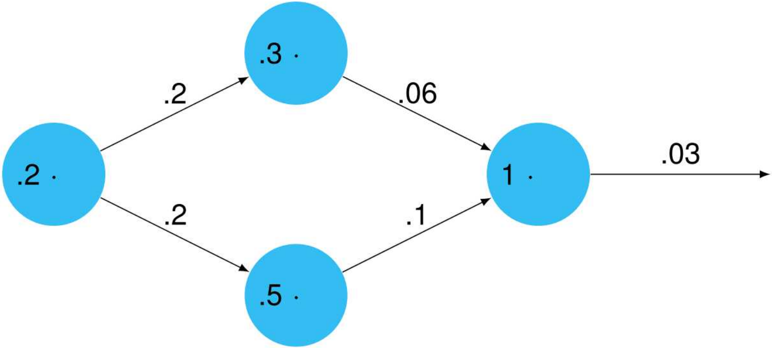
Set Up

- Cylon v1.0 is represented by a network with information processing nodes and edges and the nodes' associated weights and Boolean operators: e.g. "and", "or", "nand", "xor", etc.
- The weights represent the probability that data at the respective node transfers successfully at some time
- It is assumed that each node has a successful transfer independent of all the others
- Cylon v1.0 determines the reliability of the nodes of the network

Example

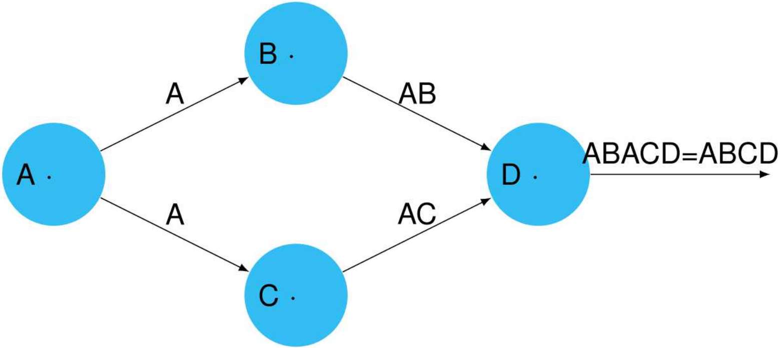


Example



Example

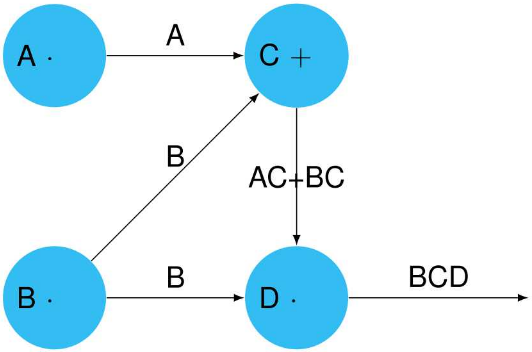
Easier to use events of success


A ·
B ·
C ·
D ·
A
A
AB
AC
ABACD=ABCD

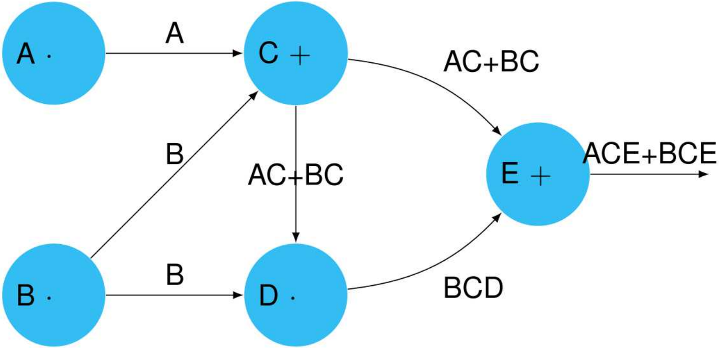
Goal

- The probability that a node is successful, depends on the data it needs to receive in order to process its computation (this probability is called the node's reliability)
- Cylon v1.0 computes the reliability for each node of a continuously projected data flow on a network at its steady state
- Given a directed graph, with no simple loops, each node is assigned a probability of functioning properly and the reliability of each node is then calculated in order of independence

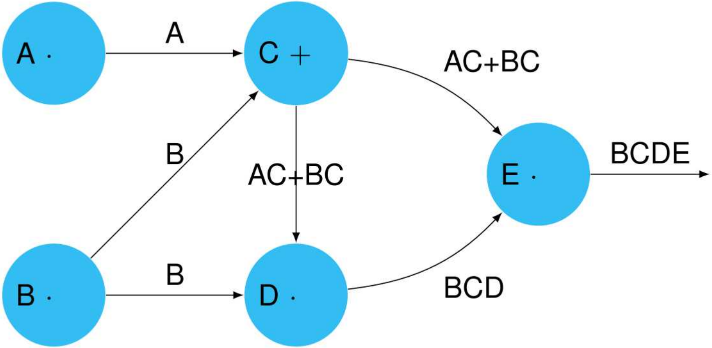
Example



Example



Example



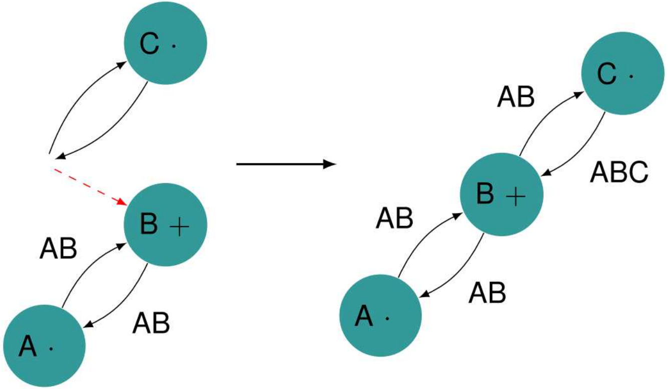
Loops

- When the possibility of simple loops are added into model, computations are not so straight forward
- The main difficulty with loops is that a node in the loop cannot simply wait for the nodes it depends on to first compute their event of reliability, because they may also be depending on the event of reliability of the node in question

This forces Cylon v1.0 to handle the loops in a separate manner, and will require justification.

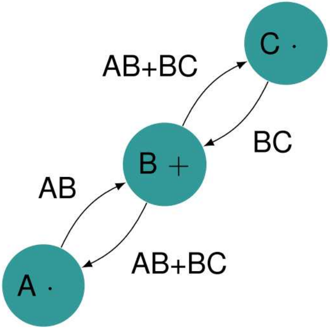
Example

Calculating one loop at a time...



Example

Calculating one node at a time...



Formal Construction

Formal definitions are required before any problems can be formally described and solved in this regime. It is assumed that the reader is familiar with networks (graphs) and weighted nodes.

Definition

Consider a directed network $(V, A, P, \odot)$ *so that each node* i *is equipped with a weight* P_i *in* $[0, 1]$ *and Boolean operation. Then this network is called a data driven network. The node's weight is known as the operational probability.*

Formal Construction

Definition

Suppose a data driven network $(V, A, P, \bigodot)$ *and probability space* $(\Omega, \mathcal{F}, \mathbb{P})$ *is given. Let* $V_i \subset V$ *be the set of nodes* j *such that the network contains the directed edge* (j, i)*. Let* $\mathcal{S}$ *be the set of sequences of nodes. For each node* i *let there be a function from sequences of nodes to* $\mathbb{P}$*-measurable subsets of* Ω*,* $E_i : \mathcal{S} \to \mathcal{F}$*. Denote the Boolean operation for each node* i *with* $i\bigodot$ *and the weights as* P_i*. Consider events* F_i *for* $i \in V$ *so that they are independent and* $\mathbb{P}(F_i) = P_i$*. Then the collection of functions* $E = \{E_i : \text{for } i \in V\}$ *is called data deducing on this network if for any sequence* $(s_1, s_2, \ldots, s_n) \in \mathcal{S}$ *with* $n \geq 1$*,*

$$E_i(s_1, s_2, \ldots, s_n) = \begin{cases} E_i(s_1, s_2, \ldots, s_{n-1}) \cdot \left({}_i\bigodot_{j \in V_i} E_j(s_1, s_2, \ldots, s_{n-1}) \right), & \text{for } i = s_n \\ E_i(s_1, s_2, \ldots, s_{n-1}), & \text{for } i \neq s_n \end{cases} \tag{1}$$

and for empty sequences $E_i() = F_i$*.*

Theorem

Suppose a data driven network is given so that the Boolean operations are all in $\{+, \cdot\}$, and the collection of functions E is data deducing on this network. It can be shown that for any sequence $(s_1, s_2, ..., s_m)$ of nodes such that each node is repeated sufficiently many times the output of the collection E has converged in the sense that for any strictly large sequence $(s_1, s_2, ..., s_n)$, $E_i(s_1, s_2, ..., s_n) = E_i(s_1, s_2, ..., s_m)$ for each $i \in V$. Moreover, the aforementioned converged events are unique over the choice of sequence. The proof for these claims comes from the monotonicity of E, the finite order of V, and the bounded $\mathbb{P}$-measure of Ω. Note that in equation (1) for $i = s_n$, the term $E_i(s_1, s_2, ..., s_{n-1})$ in the product may be replace with F_i for computing purposes, because

$$F_i \cdot \left({}_i\bigodot_{j \in V_i} E_j(s_1, s_2, ..., s_{n-1}) \right) \subseteq E_i(s_1, s_2, ..., s_{n-1}).$$

Unless otherwise specified, assume that the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ is given where $\mathbb{P}$ is a Borel measure and $\mathcal{F}$ is the power set of Ω.

Theorem

Theorem

Suppose that a data driven network $(V, A, P, \bigodot)$ *and a data deducing collection* E *are given. Suppose that the Boolean operations are all in* $\{+, \cdot\}$. *Then for any sequence s which visits each node infinitely many times there can be only one solution to Es, and there exists an integer n so that* $Es = E(s_1, s_2, ..., s_n)$.

Proof

Without loss of generality, assume $V = \{1, 2, \dots, |V|\}$, and let $V_i \subset V$ be the set of nodes j such that the network contains the directed edge (j, i). Clearly $E_i(s_1, s_2, \dots, s_{n+1}) \subseteq E_i(s_1, s_2, \dots, s_n)$, because $E_i(s_1, s_2, \dots, s_{n+1})$ is the intersection of $E_i(s_1, s_2, \dots, s_n)$ and another set.

Since the order $|V|$ is finite, there are only finite many sets F_i. Also for any sequence s, $E_i s$ is of the form $\sum_{I \in \mathcal{I}} \prod_{i \in I} F_i$ which has only finite many expressions. Indeed, $\prod_{i \in I} F_i$ has at most $|V|$ terms and so there are as many as $2^{|V|}$ expressions if the empty set were allowed. That is, $\sum_{I \in \mathcal{I}} \prod_{i \in I} F_i$ can have only $2^{|V|}$ unique terms, hence the range of E_i is finite for each i. Note that many terms need not appear in the same expression due to set containment.

By the monotone convergence theorem for nets the expression $E_i s$ converges for each i and for any sequence s. Moreover, since the range of each function E_i is finite, there exists a finite subsequence $(s_1, \dots, s_{n_i})$ of s so that $E_i(s_1, \dots, s_{n_i})$ has already converged. Finally, since there are only finite many functions E_i, there is a finite subsequence $(s_1, \dots, s_n)$ of s so that $E_i(s_1, \dots, s_n)$ has converged for each i.

Proof

Consider two sequence s and t which both visit each node infinitely often. Let $n \in \mathbb{N}$ be so that $E(s_1, s_2, \dots, s_n) = Es$ and $m \in \mathbb{N}$ be so that $E(t_1, t_2, \dots, t_m) = Et$. Suppose that $Es \neq Et$. Without loss of generality assume that there exists an $i \in V$ so that $E_i t \subsetneq E_i s$. Then there exists $k \in \mathbb{N}_0$ less than m (the sequence for $k = 0$ is interpreted as the empty sequence) so that $E_i s \subseteq E_i(t_1, t_2, \dots, t_k)$ for each i and $E_{t_{k+1}}(t_1, t_2, \dots, t_{k+1}) \subsetneq E_{t_{k+1}} s$. But

$$\begin{aligned} E_{t_{k+1}} s &= E_{t_{k+1}} s \cdot \left(\bigodot_{j \in V_{t_{k+1}}}^{t_{k+1}} E_j s \right), \text{ by equation (1)}, \\ &\subseteq E_{t_{k+1}}(t_1, t_2, \dots, t_k) \cdot \left(\bigodot_{j \in V_{t_{k+1}}}^{t_{k+1}} E_j(t_1, t_2, \dots, t_k) \right), \text{ by assumption}, \\ &= E_{t_{k+1}}(t_1, t_2, \dots, t_{k+1}), \text{ by equation (1) again.} \end{aligned} \tag{2}$$

This is a contradiction, so $Es = Et$.

Reliability

In light of Theorem 3, the following is well-defined.

Definition

Suppose that a data driven network $(V, A, P, \odot)$ and a data deducing collection E are given, with $\odot$ all in $\{+, \cdot\}$. Assume s is a sequence which visits each node infinitely many times. Then the reliability of node $i \in V$, often written R_i, is given by $R_i = \mathbb{P}(E_i s)$.

Example