Qualitative Modelling of Electrical Circuits

M H Lee *and* **A R T Ormsby**
A.I. & Robotics Research Group, Department of Computer Science,
The University College of Wales, Aberystwyth, SY23 3BZ, UK.
Email: mhl@aber.ac.uk, aro@aber.ac.uk
Telephone: +44 970 622421
Fax: +44 970 622455

Abstract

This paper presents a new model for the qualitative analysis of electrical circuit behaviour. We show that a qualitative representation of electrical resistance provides a good intuitive model of connectivity. Features include an extended qualitative symbol set for current flow and the concepts of primary and secondary levels of activity. The algorithm assigns labels to network junctions, finds current paths from source to sink, and can make predictions about the effects of circuit topology changes.

1 Introduction

This paper describes some research into intelligent reasoning tools to assist engineers in the analysis of electrical circuit behaviour. A new model to support qualitative reasoning about circuit topology and current flow is presented and a new algorithm is given for the assignment of qualitative states in electrical circuits. The class of circuits treated here cover DC electromechanical systems rather than electronics; examples are found in the physical wiring systems in avionic and automobile applications.

Our motivation is to produce tools which can assist engineers in the identification and analysis of certain classes of circuit behaviours, especially failure behaviours that have safety implications. These are very relevant in studies of hazard analysis, in the design of safety-critical systems and in many diagnosis problems. We have pursued this topic in the context of a project in Failure Mode and Effects Analysis (FMEA) for electromechanical systems [4], but the issues have wide generality and have relevance for design and diagnosis in many branches of engineering.

2 The FMEA Scenario

Consider the electrical system of an aeroplane or an automobile. The way FMEA is carried out is by proposing and answering a comprehensive series of "what if this happens" questions. The design engineer identifies a set of components that could fail and then, for each component, lists the different ways in which it could fail — this gives a set of failure modes. Then the engineer considers how each failure mode affects the other parts of the circuit and estimates the severity of the resultant behaviour. In this way, the effects of each possible fault in the set of failure modes are examined and recorded. In a great many cases, electrical faults can be expressed as either a short circuit or an open circuit. Even in more subtle failures it is often deemed satisfactory to represent the change to the system as an open or short circuit event.

Figure 1. shows an example circuit for illustration. This circuit, which is based on a fragment of an automobile cruise control system, contains typical elements found in this class of application. The switch S_1 is thrown by the user to cause the electronic control unit (ECU) to energise the relay, thus closing S_2 and completing the circuit containing the indicator bulb B. A fuse F is

provided for overload current protection and the connection $c-g$ is used by the ECU to sense the current status in the indicator bulb. Some typical possible failures for this circuit are listed below:

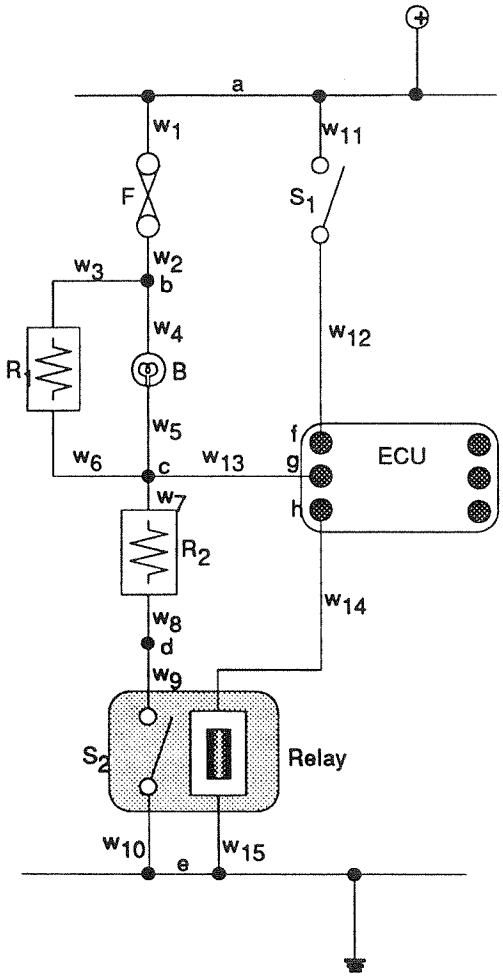

Figure 1: An example circuit

1. component B (the bulb) fails — its resistance changes to open circuit
2. the relay fails — its coil goes open circuit
3. the relay switch S_2 fails — the contacts of S_2 weld stuck at short circuit
4. the ECU fails — one of its terminals, g, acts as a short circuit to ground
5. a short circuit path occurs between points a and c — a wiring fault

In practice, to ease the problem of identifying enormous numbers of different failure cases, engineers define certain salient sets of failure types that can then be applied mechanically to all parts of the circuit, e.g. consider all the faults represented by *shorting each connection* to ground, or consider all faults caused by *breaking each wire* in turn. Further simplifications can be gained by noticing structural equivalences, for example, in a series circuit it is not necessary to break *each* wire, it is sufficient to break *any* wire.

The result of an FMEA analysis for one of these failures is a statement of the nature and criticality of its effects. For example, for the fourth of the above faults, — a short to ground

on terminal g — the resultant effect might be that circuit section a, b, c becomes activated while the ECU is rendered ineffective. Using electrical knowledge the engineer would reason that the indicator B will erroneously light and the ECU has entered a damaged or, at least, disabled state. Next, by calling on non-electrical domain expertise, the engineer could record these two effects as a minor and a major/catastrophic effect, respectively. Systems for scoring the criticality of effects have been devised and include factors for likelihood of failure, ease of detection and seriousness of the outcome. We are studying the automation of the effects evaluation stage [7] but do not include this or other risk assessment methods, e.g. FTA and HAZOP [8], in the present paper.

FMEA is a very tedious process and yet demands the expertise of a professional engineer. At present the only tools to assist in the process are bookkeeping programs that help with some of the clerical aspects. What is required are intelligent tools that can perform routine reasoning to determine the extent of electrical changes in a circuit due to postulated faults. While the judgement and domain knowledge of the skilled engineer are unlikely to be completely automated, an assistant that understands basic circuit properties and topologies and could be used to compute the effects of given faults would be of great benefit to the whole process. We are also researching into the use of these methods in diagnosis tasks, but that is reported elsewhere [5].

3 The Role of Qualitative Reasoning in Circuit Analysis

There has been some work in the past on using qualitative techniques to analyse electronic circuits. Possibly the best example is [1], where the standard ideas of electricity theory such as Kirchoff's and Ohm's laws are recast in a qualitative framework. However, emphasis in much of the previous work, and much more obviously in [2, 3, 6] has been concerned specifically with electronics.

Our work only addresses DC analysis, and has two rationales:

(a) as an experiment in qualitative reasoning, by using qualitative resistance as a new approach for modelling electrical circuit elements; and

(b) as a pragmatic approach to building front-end focussing mechanisms for circuit failure reasoning systems.

A traditional approach to circuit analysis would be to use a conventional numerical analysis package and compute the actual values of the voltages and currents for the given circuit. This would involve solving the circuit equations both before and after a proposed fault and then comparing the results to detect the changes. In addition to the overhead of having to solve the whole circuit twice every time, the determination of significant changes in numeric data is not straightforward. For a connected resistive mesh any change in value of almost any component will affect almost all the current values in the system *to some degree*. Because real numbers are being used some form of range criteria will have to be adopted to decide if the degree of change is significant. For example, in a branch directly shorted out by a wire there will always be a current, however small, as the shorting wire must have *some* resistance, (in car wiring, voltage drops along wires are significant measurements, despite being relatively large conductors). Thus, we must create change tolerance rules, e.g. currents of less than 10 milli-amps, say, are considered to be zero. However such rules will vary (a) over the circuit and, (b) with different circuit states and conditions. We may even end up with more definitions of range detection rules than components. This shifts the burden of work from numeric equation solving to the domain dependent interpretation of all the electrical changes.

Initial circuit reasoning processes are inherently *qualitative*; the engineer wants to know, as a result of a fault like those above, which parts of the circuit lose power and which parts become energised. It is usually not necessary to know the exact voltage or current values in the affected branches as it is really *the fact that changes have occurred* and *where the changes are* that is of importance, rather than the magnitudes of the changes. Non-electrical domain knowledge can then be used to decide if the change indicates a serious event. Sometimes further detailed analysis using numerical methods may be required but frequently the engineer will have sufficient knowledge to infer the consequences without finer levels of detail. Thus qualitative reasoning can be seen both as a first attempt to solve the failure analysis problem and secondly as a focusing mechanism for

partitioning the problem for optional finer scale analysis.

4 Qualitative Resistance

We adopt the conventional working approximation that any circuit can be represented by a suitable network of interconnected lumped elements, and a single two-terminal power source.

Current passes from the source terminal, called the supply, (normal convention: positive) though the resistances to the sink or ground terminal (negative).

We now define a qualitative value set for the resistance of circuit components, $[0, \ell, \infty]$. These correspond to *short circuit, load* and *open circuit.* These values relate to the engineers' intuitive notions of electrical conduits: components with resistance of value 0 pass current freely, components of resistance ∞ block all current, and components of resistance ℓ act as a significant energy absorbing load. Here 'significance' is determined by the engineers' interpretation of the application — a motor will usually have a resistance of ℓ but might sometimes be represented as 0, (if it was mechanically stalled, for example).

In this work, only resistance can be represented as an electrical property. Thus any component to be simulated must be modelled in terms of some configuration of resistances. This might seem restrictive as the energy storage properties of inductance (L) and capacitance (C) can not be modelled and so we are forced to ignore transients, time constants and all other dynamic effects. However FMEA (and much of diagnosis) is concerned with steady state analysis. This means that switches, relays and other forms of both passive and active components are handled by performing separate analyses for each state to be investigated. Consequently, sequential events involving identifiable state changes can be analysed, while short transient effects cannot.

At a qualitative level we can view resistance as a first order approximation model of electrical components. This gives a very coarse grain size but has benefits of simplicity of abstraction — such crude approximations are in the spirit of preliminary engineering analysis [9].

We notice that quite complex circuit components, for example electronic units including programmable devices like PLCs and CPUs, can be treated in this way. Providing the internal logic of the component is understood and is translated into the *electrical behaviour at the terminals for the states in question*, it is possible to reason about the gross behaviour of the fault without recourse to the internal functioning of the components. Thus a CPU can be modelled as a multi-terminal component with defined electrical relationships (a network of resistances) between the terminals for each given state of the system.

5 Component and Circuit Models

Our qualitative circuit simulator is object-oriented; all physical entities are represented as objects and connections are specified as relationships between object features. A catalogue of component models is organised in a hierarchy, e.g. "wire" is a subclass of "two-terminal-component".

All components are seen by the user as electrical "boxes" that can only connect and interact through their terminals. The catalogue stores definitions of the terminals in addition to a model of the internal function of the device. Many components are two terminal devices. These include simple electrical devices such as resistors, fuses and diodes. Most of these will be modelled as a single internal resistance. But wires and other conductors are also components, i.e. two terminal devices with a default internal model of zero resistance. Components with more than two terminals include connectors, switches, relays and various function boxes which encapsulate specific proprietary circuits. Models for multi-terminal devices will depend upon the nature of the device itself. The models are designed for normal-mode function but also take account of the need to best represent the most likely forms of fault.

Using component models we can now reduce the circuit in Figure 1 to an abstract graph. The input to our simulator consists of a list of components, with their values, and a net-list giving terminal associations. The resulting circuit model is a graph containing nodes of two classes:

component nodes, which contain qualitative resistance values, and **terminal nodes**, which define the terminal inter-connections. The circuit from Figure 1 is shown as such a graph in Figure 2.

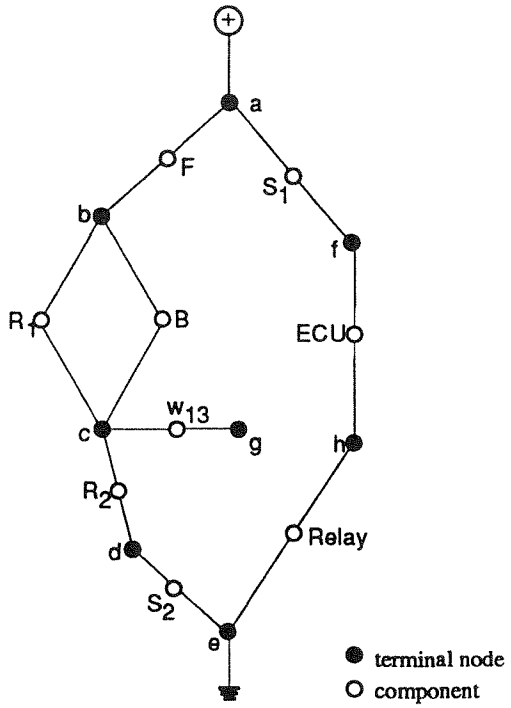

Figure 2: The circuit model

Although wires are two-terminal components and are included in the analysis, we eliminate them from the diagrams for simplicity. Also, all analysis takes place on the terminal nodes and so the component values can be treated as weighted edges. Notice how, in this example, the ECU is modelled as a direct connection between f and h, with no internal connection to g.

6 Qualitative Composition Functions

In order to determine the resultant value of aggregated components we must define an algebra for qualitative resistance. For the common topologies of series and parallel connection schemes we recognise two-terminal sub-networks, called *chains*, and *bunches*, respectively.

We define an ordering for qualitative resistance: $0 < \ell < \infty$. The physics of serial aggregation in chains require resistance values to be summed, and we observe this is satisfied by taking the maximum of the qualitative values. Similarly for bunches, which normally are reduced by the summation of the constituent conductances (the reciprocal of resistance), the minimum value of the constituents gives the result of the qualitative combination. These rules for networks of chains and bunches are summarised in Table 1.

By repeatedly applying the aggregation rules we may reduce any circuit down to a simpler equivalent circuit. If there are no bridge constructions, the final result will be a single resistance and its value will indicate the electrical state of the network. Thus, if the final value is ∞ there can be no current flowing and the entire circuit is inactive, if the final value is 0 there is a short circuit condition between the power supply terminals, and if the value is ℓ then at least part of the circuit is active and drawing current.

We have implemented chain and bunch finding algorithms which perform reduction of the circuit and we notice that such global data is valuable as first level information about circuit

network A	network B	A & B chained	A & B bunched
∞	∞	∞	∞
∞	ℓ	∞	ℓ
∞	0	∞	0
0	∞	∞	0
0	ℓ	ℓ	0
0	0	0	0
ℓ	∞	∞	ℓ
ℓ	ℓ	ℓ	ℓ
ℓ	0	ℓ	0

Table 1: Aggregation algebra for chains and bunches; max R and min R

behaviour. However, finer detail is required — we would like to know *which* parts of the circuit are active and *which* parts are inactive, also we would like to find the components that are shorted out and those that are on power short paths. In order to assign qualitative status values to individual components we have developed algorithms that use the above algebra for reducing chains and bunches but reflects the results of these aggregations so that each constituent part receives a value for its local condition. More importantly, this method also deals with those bridging circuits that can not be handled as combinations of chains and bunches.

7 Qualitative Circuit Status

Using the methods to be described, each component will be assigned an indicator that reflects its activity. There are two types of status indicator corresponding to qualitative current and voltage. Current can only be assigned to components, i.e. edges in the graph, while voltage can only have meaning at the terminals, i.e. the nodes.

7.1 The current status indicator

This is the variable I and is assigned one of $[O, S, P, \Uparrow, A]$ which have the following meanings:

O Open circuit. No current flowing as there is no active path that passes through this component.

S Short circuit. No current flowing as some path(s) short out this component.

P Power short. This component is on a direct short path between the supply terminals. The total path resistance has been determined as 0 as there are no *load* nodes on the path, and overload current will flow. This obviously represents a major fault condition.

$\Uparrow$ Load path. The component carries load current and forms part of an active branch. The direction gives the convention for expected flow from positive to negative.

A Ambiguous flow. Both terminals of this component have similar values and so either direction of flow is possible. The flow is likely to be smaller than in the above case and, if the circuit was balanced, could even be zero.

7.2 The voltage status indicator

This variable, V, takes a value from $[+, 0, -, \sim]$. Parity with one of the power terminals is signified by $+$ or $-$; this can occur either through a direct connection to a supply terminal or in an inactive branch where all nodes can be at the supply level. The symbol 0 indicates an inactive node with no connection to either positive or negative terminals. The symbol $\sim$ shows that the voltage lies somewhere between the supply potentials.

It can be seen that current status values of O or S correspond to I = 0, i.e. no flow in the labelled component. The voltage label supplements the current indicator by supplying information useful for predicting current flow in the event of changes to the circuit topology. For example, if the current status = O then V suggests different conditions for flow:

If
- $V = +$ a connection to $\ominus$ is required to obtain flow
- $V = -$ a connection to $\oplus$ is required to obtain flow
- $V = 0$ more than one connection is required to obtain flow
- $V = \sim$ any new connection is likely to cause flow

Likewise, if $I = S$ then:

- $V \neq \sim$ indicates that removing the short from the node will not be enough to cause current to flow through it.

These qualitative indicators are one of the main features of our approach. They provide relevant and significant information about the circuit in an intuitive form; in contrast with numerical simulators this is a much more meaningful and 'user friendly' output. We now describe the methods for computing these status values.

8 The CIRQ1 Simulator Algorithm

This section describes the basic algorithms that label network terminals with path resistance values, find paths from power source to sink, and assign current and voltage status values.

First, we define path resistance:

Definition— The **path resistance** (PR) between two nodes, s and t, is the resultant resistance calculated by using a given aggregation algebra applied to all chains and bunches that exist on all paths from s to t.

The first algorithm, *label-node-resistances*, calculates and labels all the terminal nodes in the network with their PRs in both directions, from the supply and from the ground terminals. Each node is assigned a pair of variables for storing these *forward* and *reverse* path resistances, known as f/r. The algorithm, which is a modified version of Dijkstra's shortest distance algorithm [11], begins from the positive terminal and assigns the path resistance from the supply to each node's forward label, then the reverse labels are assigned by a repeat process starting from the negative terminal. We arrange to initialise all values to ∞/∞, so that any disconnected sub-circuits that can not be reached from the supply terminals will be returned with appropriate values.

Figure 3 shows the circuit of Figure 2 after processing. It is useful to treat separately any complete branches in parallel with the power supply. In this example switch S_1 is open and S_2 is closed. This shows how the f/r values can be used to infer the activity status of the circuit components. Any node which has infinity in both forward and reverse labels is not connected to either of the terminals and hence has zero current flowing. Thus these nodes can be assigned zero for both voltage and current. Nodes which have an infinity value for one of the PR labels also have no current flowing, but have a connection to one of the terminals. All other assignments will require labels for current flow to be assigned by the flow path finder.

The *flow-path-finder* algorithm then marks all active flow paths from supply to ground. The first stage is trivial and labels the obviously inactive nodes as described above. Then, starting at the positive terminal, a path is marked through to the negative terminal. If any of the initial arcs from the positive terminal have the labelling zero/zero then there is a direct short across the power wires and this can be reported immediately. The scan follows the line of 'least resistance'; in this way the main flow paths are discovered first, followed by the more circuitous routes. The nodes are tested to detect any shorted branches, i.e. those branches with a non-zero resistance but with the same f/r values at the end nodes, s and t. Any 'dead-ends' are also detected by finding paths

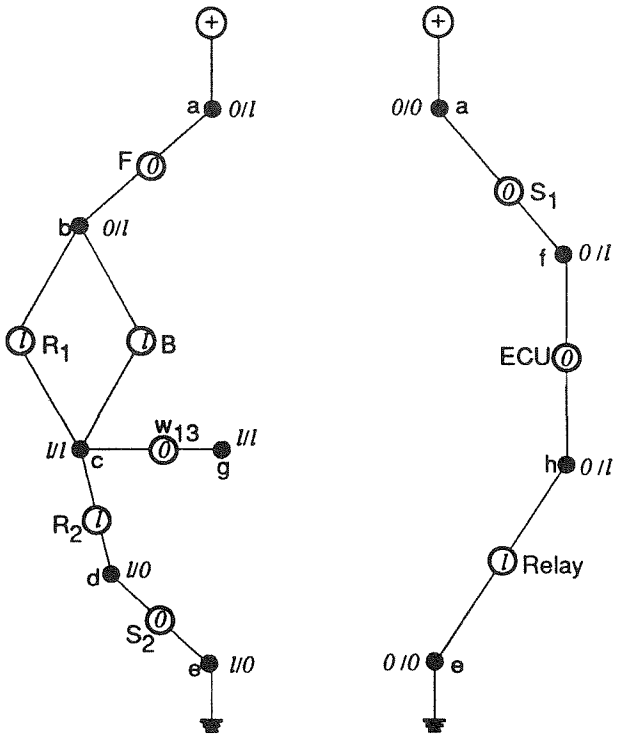

Figure 3: Path labels for CIRQ1. Components have qualitative resistance values. Terminals are labelled with f/r values.

or branches which have no unmarked exit route. All active current nodes are assigned voltage labels of $\sim$.

9 An Augmented Algorithm: CIRQ2

The CIRQ1 algorithm contains the essence of qualitative reasoning for resistive circuits. While nothing that follows invalidates the above method, we have found that slightly different aggregation functions can produce considerably more interesting output. It is important to realise that variations which augment the labelling of the ℓ values can be devised that will not affect the results, *providing* that the ordering $0 < \ell < \infty$ is maintained.

Our augmentation is to extend the f/r labels to allow *positive integers* to represent the results of the aggregation algebra. Then any path value represents either open circuit, 0, short-circuit, ∞, or some integer, n, and the relation $0 < n < \infty$ still holds. Now any branch with an integer label is considered equivalent to the previous label ℓ, but two branches with labels, n and m, can be combined by any function that maps into another positive integer. Aggregation functions which satisfy this condition include: sum, max, min, etc.

The algebra we have found most useful is summation for serial chains and minimum for bunches. Table 2 shows this scheme.

The algorithm for f/r calculation is virtually as before, but the aggregation algebra is modified as in table 2. The f/r values now represent resistive distances from the supply and ground respectively, *in terms of the number of loads*; they begin at zero at the supply and ground nodes and increase by one for each load value further away.

Figure 4 shows the values for our example. The labelling at the positive supply node, n/m, is reflected at the negative node as m/n. If we define the *total path resistance* (tpr) as $n + m$ then there will be at least one path for which $f(n_i) + r(n_i) = tpr$, for all nodes n_i on the path. We

chain A	chain B	A & B chained	A & B bunched
∞	∞	∞	∞
∞	m	∞	m
∞	0	∞	0
0	∞	∞	0
0	m	m	0
0	0	0	0
n	∞	∞	n
n	m	$n+m$	$min(n,m)$
n	0	n	0

Table 2: Aggregation using Sum R and Min R

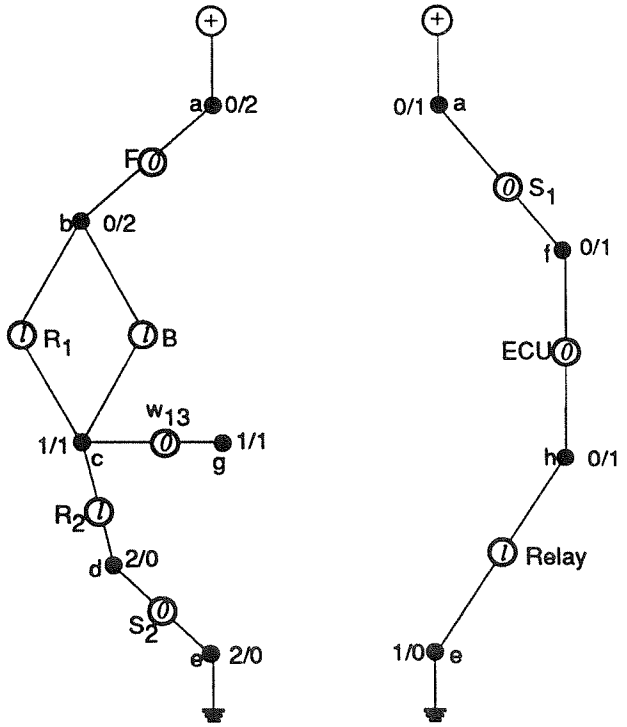

Figure 4: Path labels for CIRQ2. (S_1 and S_2 closed).

term these paths, where all the nodes have the same *tpr* value, the *primary* paths. We denote the *tpr* of primary paths, *PPR*, (Primary Path Resistance).

Primary paths can be summarised as follows:

> If the positive supply node is labelled *0/m*, where m is a positive integer, then at least one primary path must exist, from positive to negative, through a sequence of active nodes where, for each node, $tpr = m$. No short circuited and open circuited nodes are found on primary paths; these can only occur on *secondary* paths which have $tpr > m$.

This concept of primary and secondary paths is important and very useful for our purposes. By segregating the circuit into two parts — (a) known paths in which all nodes are active and (b) other paths that include all shorts, open circuits and other less direct routes — we can tackle analysis problems in two stages.

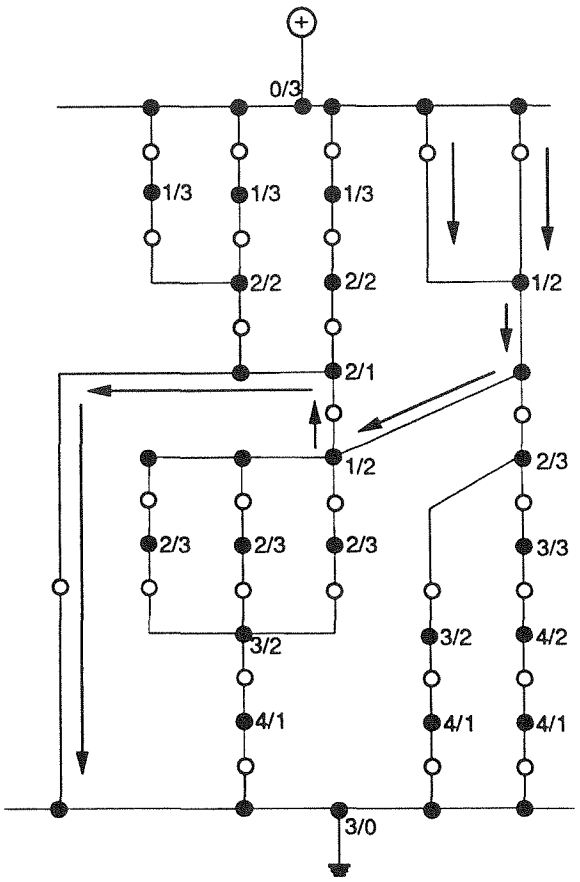

Figure 5: Path finding in CIRQ2. Primary paths are marked by arrows.

Figure 5 shows a more complex example; all component values are ℓ. This clearly shows how the primary paths, with $tpr = 3$, identify the "paths with the minimum number of load resistances".

10 Further Aggregation Algebras

The graph-theoretic literature contains many shortest path methods, e.g. Ford [12] has dealt with negative valued arcs, Pollack [13] has a method for second-shortest paths, Minieka [14] can find the kth shortest path, and Floyd's [15] algorithm finds all shortest paths. However by pursuing out qualitative scheme we discover several other aggregation functions that provide results at least as effective as graph theoretic methods.

We notice that by considering conductance rather than resistance we can find dual circuits. If a component has qualitative resistance R then its qualitative conductance is G, as follows:

R	0	ℓ	∞
G	∞	ℓ	0

Table 3: Resistance and Conductance

For consistency, we show component values as resistances in all the following tables, although the results have been derived by considering conductance values. We can now apply summation R and Maximum G to chains and bunches respectively, and obtain the following rules:

chain A	chain B	A & B chained	A & B bunched
∞	∞	∞	∞
∞	m	∞	m
∞	0	∞	0
0	∞	∞	0
0	m	m	0
0	0	0	0
n	∞	∞	n
n	m	$n+m$	$max(n,m)$
n	0	n	0

Table 4: Aggregation using Sum R and Max G

Applying this algebra to the previous example gives the results shown in Figure 6. This version identifies the "paths with the maximum number of load resistances". The *tpr* of 9 at the supply can be followed to locate the nodes on the 'longest' path.

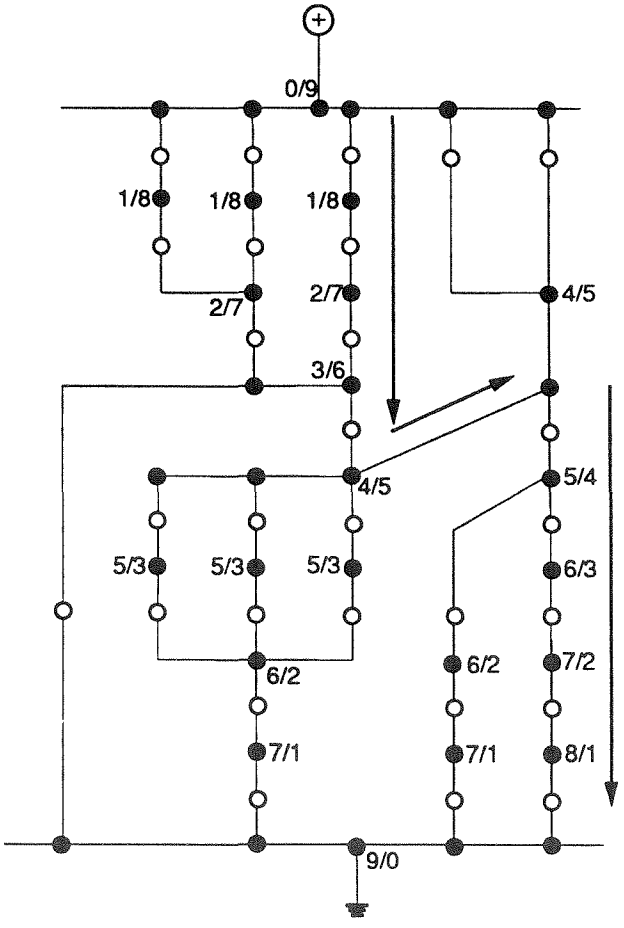

Figure 6: Path labels for sum R, max G. Longest load path marked by arrows.

Another interesting case occurs when using minimum G for chains and summation G for bunches, as seen in table 5.

An example of this algebra is shown in Figure 7. This case gives the "the number of paths

chain A	chain B	A & B chained	A & B bunched
∞	∞	∞	∞
∞	m	∞	m
∞	0	∞	0
0	∞	∞	0
0	m	m	0
0	0	0	0
n	∞	∞	n
n	m	$min(n, m)$	$n + m$
n	0	n	0

Table 5: Aggregation using Min G and Sum G

leaving a node on route to a supply terminal". If a node has f/r values of n/m then there are n paths from this node to the positive supply and m paths to the negative. Hence, this explains why the supply terminals have different values. This is related to the minimum cut-set needed to deactivate the circuit.

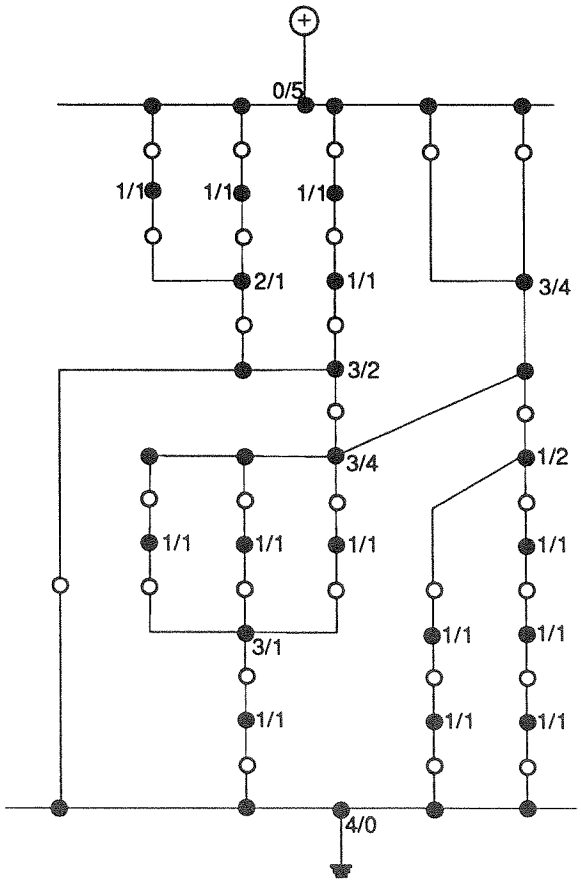

Figure 7: Path labels for min G, sum G.

Finally, the dual of table 5 gives maximum R for chains and summation G for bunches, giving table 6.

Figure 8 shows an example of this last algebra applied to our circuit. Now the results give the

chain A	chain B	A & B chained	A & B bunched
∞	∞	∞	∞
∞	m	∞	m
∞	0	∞	0
0	∞	∞	0
0	m	m	0
0	0	0	0
n	∞	∞	n
n	m	$max(n, m)$	$n + m$
n	0	n	0

Table 6: Aggregation using Max R and Sum G

"number of distinct (exclusive) paths from a node to the supply". This produces large numbers due to the combinations of parallel circuits.

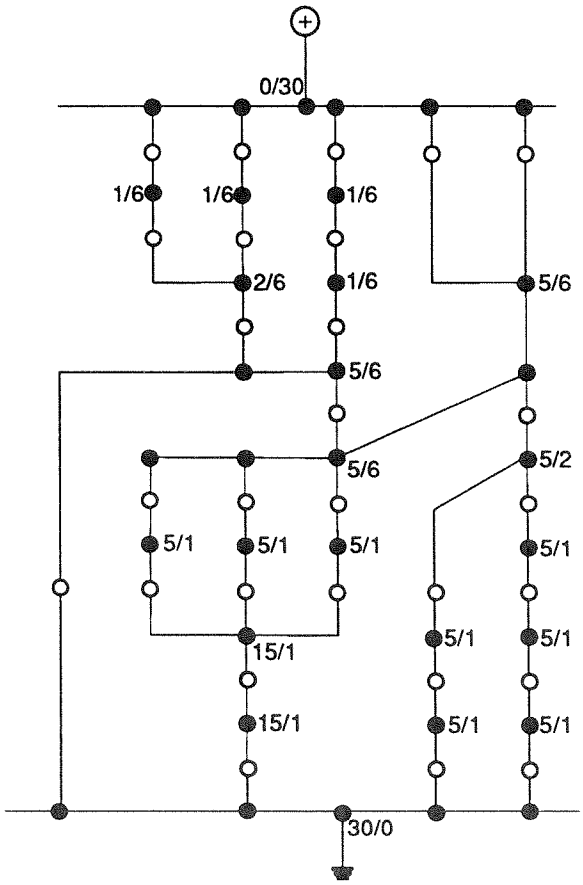

Figure 8: Path labels for max R, sum G.

11 Summary

These results are properties of the graph model which reflect relevant topological features of the electrical circuit. The first two are "depth properties" in that they measure maximum and

minimum path lengths between the supply points. The last two are "breadth properties" in that they record the maximum and minimum "widths" of the circuit.

These results are produced as a product of the scanning algorithm ($O(n^2)$) rather than by graph-theoretic methods and they may all be calculated and stored concurrently during the processing of a particular circuit. We are aware of other techniques that could be used, such as network flow algorithms, e.g. in [12, 14] and behaviour based aggregation ideas [10], but we argue that our method is more efficient in operation than numerical methods and is better tuned to the intuitive notions of electrical circuits than other symbolic representations.

The different composition functions have different benefits for specific applications. We have found the Sum R, Min R algebra most useful, mainly because of its ability to separate out primary and secondary paths in a way that is suitable for FMEA. However, the path with the maximum number of active load nodes may be of interest when diagnosing a faulty active circuit, and both the minimum number and the maximum number of paths between selected nodes flow may be important in cases of network analysis, e.g. for redundant routing when sections are disrupted.

The qualitative representation of electrical resistance presented here is an intuitive model of connectivity. We realise that the labels are really a local indication of circuit topology. Thus, the value ∞ at a node records that the node is disconnected from some other (implied) reference point. The value ℓ tells us that the node is connected by some path to the reference and the value 0 indicates that the node is, at least electrically, identical to the reference point. Hence, this scheme can be seen as a non-graphical method for reasoning about the connection topology of a circuit. We also see why many variations of the aggregation algebra are possible and why these don't affect the basic results *providing* the ordering $0 < \ell < \infty$ is maintained. This is because the numeric labelling schemes are cleanly superimposed on top of the fundamental symbolic representation and do not perturb any of its (coarser) results.

Regarding our FMEA application, it is most important that gross changes in electrical activity, due to given faults, (i.e. short and open circuits), are located quickly and effectively. Our system represents these features in a direct and simple manner and provides a layered approach to the computation of electrical status. The qualitative technique described in this paper offers a contribution towards automated tools that model some of the intuitions used by domain engineers. The method also provides a framework for generating predictive information, such as the likelihood of current flow for a given change to a circuit.

The present algorithm is being used in further research on tools to support engineers in FMEA and diagnosis tasks on electrical circuits.

References

[1] Johan de Kleer. How Circuits Work. In Daniel G Bobrow, editor, *Qualitative Reasoning about Physical Systems*, pages 205–280. MIT Press, 1985.

[2] G J Sussman. Electrical design: A problem for artificial intelligence research. In *Proceedings IJCAI-77, Cambridge, MA*, pages 894–900, 1977.

[3] Brian C Williams. Qualitative Analysis of MOS Circuits. In Daniel G Bobrow, editor, *Qualitative Reasoning about Physical Systems*, pages 281–346. MIT Press, 1985.

[4] Price C.J., Hunt J.E., Lee M.H. and Ormsby A.R.T., 'A Model-Based Approach to the Automation of Failure Mode Effects Analysis for Design', Proc. Institute of Mechanical Engineers, Part D: The Journal of Automobile Engineering, vol 206, pp7, 1992.

[5] Hunt J.E. and Price C.J. 'An Augmented Model-Based Diagnostic System Exploiting Diagnostic and Domain Knowledge', in "Research and Development in Expert Systems VIII", edited I. Graham and R. Milne, pp25-39, CUP 1991.

[6] Harry G Barrow. VERIFY: A program for proving correctness of digital hardware designs. In Daniel G Bobrow, editor, *Qualitative Reasoning about Physical Systems*, pages 205–280. MIT Press, 1985.

[7] Hunt J. E., 'Calculating the Risk Priority Number', JD035 project document, Computer Science Department, University College of Wales, 1991.

[8] Terry G. J., 'Engineering System Safety', Mechanical Engineering Publications, London, 1991.

[9] Inglis C. 'Modern Engineering', Oxford U P, 1955.

[10] Bylander T., 'A Theory of Consolidation for Reasoning about Devices', Int. J. Man-Machine Studies, 35, pp467-489, 1991.

[11] Dijkstra, E. W. 'A Note on Two Problems in Connection with Graphs", *Numerische Math.*, 1, pp269-71, 1959.

[12] Ford L. R., 'Network Flow Theory', Rand Corp., 1956.

[13] Pollack M. 'The kth Best Route through a Network', Operations Research, 9, pp578-580, 1961.

[14] Minieka E., 'Optimization Algorithms for Networks and Graphs', Dekker, 1978.

[15] Floyd R. W., 'Algorithm 97 (Shortest Path), CACM, 5, p345, 1962.