Qualitative Phasor Analysis

Juan Flores and **Art Farley**
Computer and Information Science Department
University of Oregon
Eugene, Oregon, 97403
phone (541) 346-4416 fax (541) 683-5373
email juan@cs.uoregon.edu

Abstract

Most of the work on behavior prediction on the field of Qualitative Physics has focused on transient behavior and responses to perturbations (de Kleer & Brown 1984; Forbus 1984; Kuipers 1985; Williams 1984); very little has been done about behavior of systems in steady state (Sussman & Steele 1980). An understanding of the sinusoidal steady state of electrical circuits is important for several reasons. A large class of devices and networks, especially those in the area of power generation, transmission, and distribution, are designed for sinusoidal steady-state operation (Fitzgerald 1945; Grainger & Stevenson 1994; Gönen 1988).

This paper presents a framework to reason about linear electrical circuits in sinusoidal steady state. This approach constitutes a qualitative version of *Phasor Analysis*, so it is called Qualitative Phasor Analysis (QPA).

Introduction

One of the main objectives of qualitative physics is to derive the behavior of a system from a description of its components and their interrelationships (de Kleer & Brown 1984; Forbus 1984; Kuipers 1985; Williams 1984). Prediction of behavior has been achieved by traditional physics through numerical approaches. In particular, in the area of circuit analysis (Kerr 1977) there exist a number of numerical methods to analyze circuits of different kinds and under different conditions. These methods take as input a circuit topology and exact values for the parameters, perform some computation (mainly based on linear algebra or iterative methods to solve non-linear or differential equations), and return exact values for the variables representing the unknown quantities. In all this process, causality and explanation is discarded for the sake of precision.

This paper presents a framework for qualitative analysis of linear electrical circuits in sinusoidal steady state. We call this approach Qualitative Phasor Analysis (QPA). QPA takes as input a description of circuit components and their connections, with qualitative values and constraints on the circuit parameters. Values of the parameters can be intervals, defaulting to $(0, \infty)$ (i.e. all parameters are positive). Constraints on parameters can be provided by the user; for instance, in a circuit involving two resistors, the user can assert $(< R_1\ R_2)$, or $R_1 = [5, 10]$, etc.

QPA will compute as much information as possible and return it to the user in the form of new values for circuit variables and new constraints. The results provide the user with information about the consequences of a given constraint in the circuit, A set of constraints represents a set of predicted behaviors. A primary difference from circuit simulators is that they require precise values for all parameters and return precise values for all of the circuit variables. QPA can produce results even if we do not provide any information about the circuit parameters; of course, the more specific the input is, the more specific the output will be. If all the parameters are precisely specified, the results of QPA will be like those of conventional circuit simulators (i.e. only one behavior is predicted).

QPA is based on a constraint analysis approach to qualitative physics. First, it develops a constraint-based model of a circuit, where the constraints are derived from general knowledge of circuit theory and the circuit's topology. Second, it is able to propagate constraints of different kinds; we have different sets of constraints, ranging from confluences, ordering constraints, order of magnitude relations, and phase angle constraints. We extend the previous qualitative approaches to circuit analysis to deal not only with scalar magnitudes, but also to include phase angle information.

This article explains the basic problems this research project addresses, identifies some of the important research issues, and discusses implementation and evaluation plans. The section *QPA*, presents an overview of our main circuit analysis engine, called Qualitative Phasor Analysis (QPA). The section *An Application*

Domain presents a field of application for QPA: Power Systems Analysis and Design. Section *Implementation and Evaluation* gives an overview of the system architecture and implementation details. Section *Related Work* reviews previous work published in the field of qualitative physics related to this research project. Finally, the *Conclusion* discusses the contributions and limitations of this project, as well as directions for future research.

QPA

The electrical engineering community has been very successful in predicting behavior of linear circuits in steady state. The main tool they use in circuit analysis is the phasor. Phasors (Kerr 1977, chapter 5) are a mathematical transformation that maps sinusoids from the time domain to the frequency domain, allowing us to replace complicated simultaneous differential equations (Boyce & DiPrima 1969) by algebraic simultaneous equations in the complex domain. Besides their power to solve linear circuits, phasors can be expressed in an intuitive graphical form; as *phasor diagrams.* These diagrams allow electrical engineers to have a better understanding of what happens inside a circuit and can be used to produce causal explanation of physical phenomena.

In a circuit excited by a sinusoidal voltage source, of frequency ω, all variables are also sinusoidals oscillating at the same frequency. Each variable $V(t)$ can be expressed as the real part of a complex quantity. That is

$$\begin{aligned} V(t) &= Re(|V|(\cos(\omega t + \angle V) + j\sin(\omega t + \angle V))) \\ &= |V|\cos(\omega t + \angle V) \end{aligned}$$

where $V(t)$ represents a (real) function of time, and V represents its corresponding phasor in the frequency domain. If we represent all variables in a circuit by a phasor, they will rotate at the same angular frequency as if fastened together. So a phasor diagram can be seen, at any given moment, as a snapshot of the set of rotating phasors that represent all the quantities in the circuit. The solution to the circuit can be obtained by taking the real part of each phasor (i.e. make all phasors rotate at the same frequency as the source and take each phasor's projection over the real axis). For example, figure 1 and figure 2 show a series RLC circuit, excited by a sinusoidal voltage source, and its phasor diagram, respectively.

A mathematical model of a circuit includes a set of algebraic relations that constrain its behavior. For instance, we know that current and voltage are in phase in a resistor or that the currents of two parallel branches add to the total current of the combination.

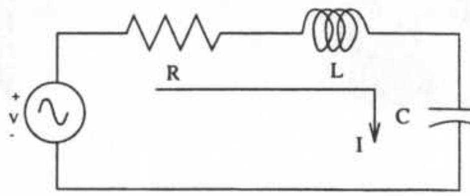

Figure 1: Series RLC circuit

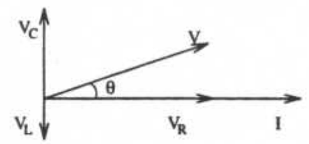

Figure 2: Phasor diagram for series RLC circuit

We can capture this information as a set of qualitative constraints that will enable us to reason about a circuit's behavior.

To determine the set of algebraic constraints, we represent the circuit as a structure of parallel/series clusters. We can recursively traverse that clustering structure, generating constraints for each cluster or component we encounter (Liu 1991). The set of constraints can be partitioned into subsets of several types: Definitions (e.g. $(DEFINITION(= V_{R_2}\ (*\ Z_{R_2}\ I_{R_2}))))$, Order of Magnitude (e.g. $(>\ I_{R1}\ I_C)$ or $(\gg I_{R1}\ I_C))$, Phase Angle (e.g. $(IN\_PHASE\ V_{R_2}\ I_{R_2})$ or $(VALUE\ (ANGLE\ I_C\ V_C)\ 90))$, and Confluences (e.g. $(CONFLUENCE\ (+\ (\partial\, VR))\ (-\ (\partial\, ZR))\ (-\ (\partial\, IR))))$.

Confluences represent constraints about change (de Kleer & Brown 1984). For instance, for Ohm's law in a resistor $V_R = Z_R I_R$, we have the qualitative counterpart $\partial V_R - \partial Z_R - \partial I_R = 0$ (represented in figure 3 in prefix form and obviating equality to zero). This confluence indicates, for example, that if Z_R decreases and V_R does not change, I_R increase.

To illustrate how the set of constraints for a given circuit is generated, let us consider the circuit of figure 3, which shows a circuit's topology as configured in terms of series parallel clusters. Figure 3 also shows examples of the kind of constraints QPA will generate for each component and cluster of the circuit. The resulting constraints constitute what we call the Basic Set of Constraints (BSOC). Once a BSOC has been generated, propagation is used to obtain the transitive closure of the constraints and their implications. For instance, from constraints $(=\ I_{S_1}\ I_{R_1})$ and $(=\ I_{S_1}\ I_L)$ we can derive $(=\ I_{R_1}\ I_L)$.

If the user has further information about features of the circuit, these can be expressed as additional constraints. For instance, the user can tell QPA

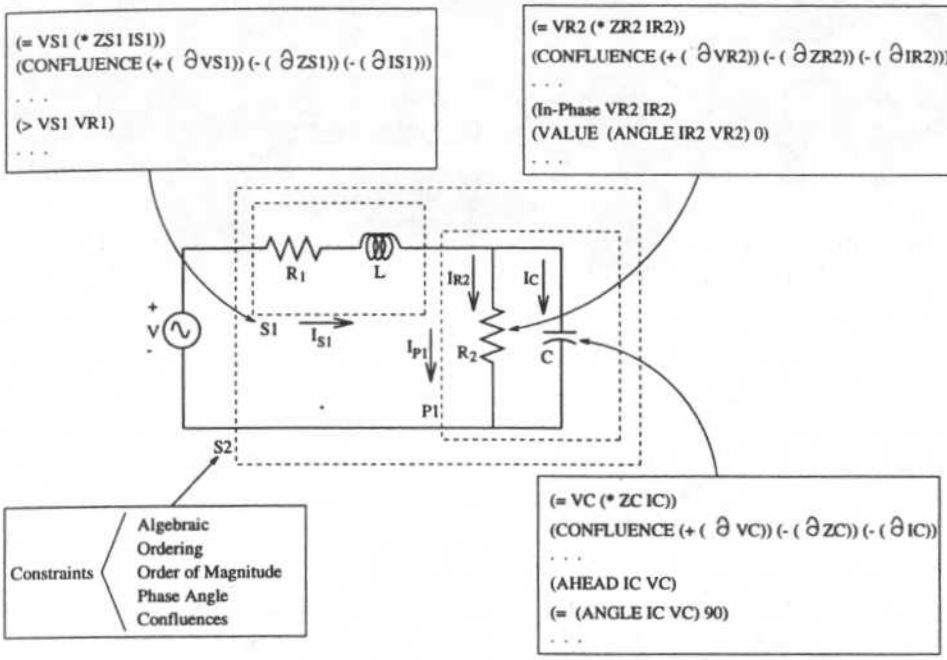

Figure 3: An electrical circuit, its configuration, and constraints

that ($<$ Z_{R_2} Z_C); propagation will indicate any implied constraints, such as ($<$ I_C I_{R_2}). The user can also ask if a certain property holds. For example, if the user asks if $(ANGLE\ I_{P_1}\ V_{P_1})$ can be 90 degrees, the system can respond with the following answer "No. You told me that ($<$ Z_{R_2} Z_C), which implies that ($<$ I_C I_{R_2}), and therefore $(VALUE\ (ANGLE\ I_{P_1}\ V_{P_1})\ (0\ 45))$."

Constraint propagation must take place across algebraic constraints. For example, for a parallel cluster like the one on figure 4, we have $(=\ V\ (*\ I_a\ Z_a))$ and $(=\ V\ (*\ I_b\ Z_b))$. If the user has provided the constraint ($<$ Z_a Z_b), we can then conclude ($>$ I_a I_b).

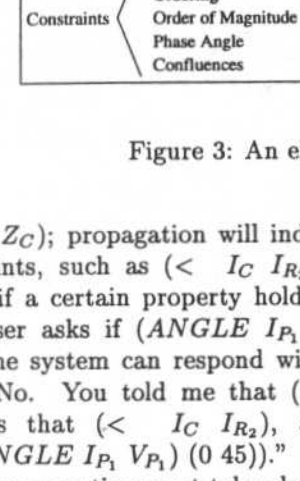

Figure 4: Two resistors in parallel

An important aspect of constraint propagation is the interaction between magnitude and phase angle variables. For simple elements, the phase angle is precisely defined, but when components of different kinds are combined, phasor addition may be ambiguous. For instance, consider a resistor in series with a capacitor, as shown in figure 5 (the phasor diagram is also shown in the figure). Depending on the rela-

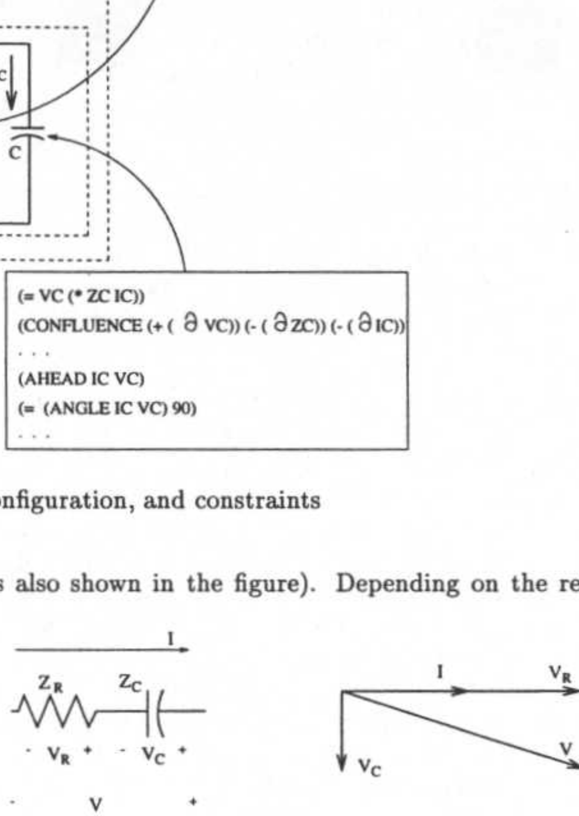

Figure 5: A resistor and a capacitor in series

tion between the magnitudes of V_R and V_C, the angle $(ANGLE\ I\ V_C)$ can have different values. If ($>$ V_R V) then $(VALUE\ (ANGLE\ I\ V)\ (0, 45))$; if ($=$ V_R V) then $(VALUE\ (ANGLE\ I\ V)\ 45)$; else if ($<$ V_R V) then $(VALUE\ (ANGLE\ I\ V)\ (45, 90))$.

The constraint propagation mechanism of QPA allows us to deal with symbolic, uncertain, or numeric values for the parameters and variables of the system. All values are represented as intervals: numbers are punctual intervals (e.g. $5 = [5, 5]$), uncertain values are intervals with open or closed limits, and symbolic values are translated into intervals as well (e.g. *positive* $= (0, \infty)$). The part of our work that deals with interval propagation, although developed independently, is consistent with the work of (Hyvönen 1989). Furthermore, it integrates value propagation with order propagation. For instance, given constraints $X = [0, 10]$, $Y = [5, 15]$, and $(=\ X\ Y)$, we can *refine* the values of X and Y to be both equal to $[5, 10]$. This feature allows QPA's solutions default to the solutions of conventional circuit solvers in the case all values are precisely defined.

Causal (First Order) reasoning is also important if we want to be able to explain changes in a circuit's behavior. Confluences capture the interaction among different variables in the circuit and how changes in one variable can produce changes in other variables. For example, if the user asks "what happens if R_2 increases?" (expressed as the constraint $(VALUE\ (\partial\ R_2)\ +))$, the system replies

> "If R_2 increases, I_{R_2} decreases, which causes I_{S_1}'s magnitude to decrease and I_{S_1}'s phase angle to increase. This decrease in I_{S_1} would cause a decrease in V_{R_1} and V_L, and therefore, in V_{S_1} and V_{S_2} (relative to V_{P_1}, which is taken as a reference for this example)".

The BSOC constitutes a partially constrained model of the circuit, which corresponds to a set of circuit behaviors; the more constrained the circuit model is, the more reduced the set of possible behaviors is. For instance, before asserting $(<\ I_C\ I_{R_2})$, we can tell that the phase angle between I_{S_1} and V_{P_1} (denoted by $(ANGLE\ I_{S_1}\ V_{P_1}))$ lies in the interval (0, 90). After asserting the above constraint, we know that $(VALUE\ (ANGLE\ I_{S_1}\ V_{P_1})\ (0\ 45))$. To have a better idea of what this set of constraints represents, figure 6 shows one phasor diagram, of the many possible, for the Circuit Model of figure 3. That phasor diagram was drawn under the added assumptions $(>\ I_{R_2}\ I_C)$, $(>\ V_L\ V_{R_1})$, and $(>\ V_{S_1}\ V_{P_1})$.

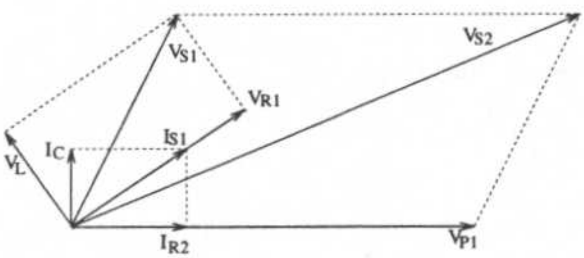

Figure 6: A possible phasor diagram for circuit in previous figure

After the user provides a number of constraints, it is more likely that further constraints are rejected as being inconsistent with the partial solution. At that point the user can say "OK, give me all possible, fully constrained models[1] ...". One of the goals of QPA is to produce all possible, fully constrained models of the circuit (i.e. all possible behaviors of the circuit under the actual set of constraints). Traversing the circuit's structure, we determine what variables are interrelated and produce all relevant constraints, branching on each possibility. Constraints that produce contradictions, are pruned and not included in the behavior tree. The result is a tree like the one shown in figure 7, where the leaves correspond to sets of constraints representing a fully constrained model of the circuit.

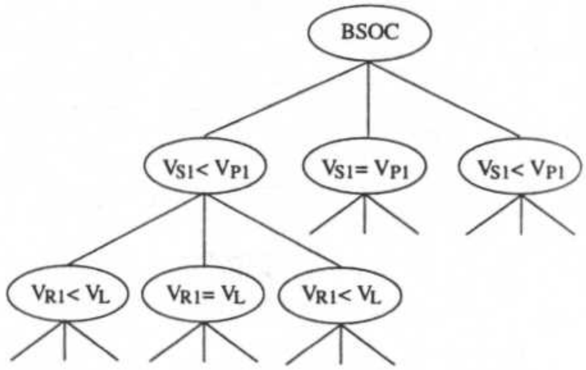

Figure 7: Tree containing all fully constrained models.

QPA handles order of magnitude constraints. Order of magnitude constraints can be used to simplify a circuit, when appropriate. Returning to the circuit shown in figure 3, if the user tells the system that $(\gg\ Z_C\ Z_{R_2})$, the system responds that $(\approx\ I_C\ 0)$ is implied. This is interpreted by QPA in the appropriate way; the current through that branch is negligible, therefore, the whole branch can be omitted (open circuited). The resulting circuit is shown in figure 8. In

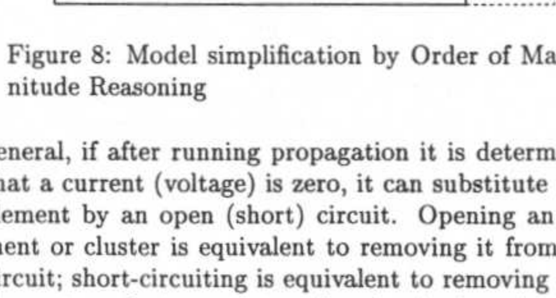

Figure 8: Model simplification by Order of Magnitude Reasoning

general, if after running propagation it is determined that a current (voltage) is zero, it can substitute that element by an open (short) circuit. Opening an element or cluster is equivalent to removing it from the circuit; short-circuiting is equivalent to removing that element or cluster and to collapsing both nodes into

[1] A fully constrained model contains an order constraint between every pair of comparable variables of the circuit.

one. After these structural modifications, a new model of the circuit is rebuilt, and propagation on the given constraints must be recomputed.

A basic form of problem solving made possible by constraint propagation is diagnosis. Consider the process of measurement interpretation or diagnosis, based on a QPA representation. Suppose that the observed state of the circuit is the one shown in figure 9. The observed state, mea-

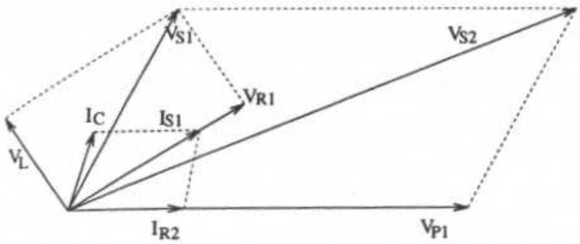

Figure 9: Faulty observed behavior

sured by physical instruments, can be easily translated to a set of constraints. For instance, from figure 9 we observe that $(= I_{R_1}\ I_{S_1})$, $(< I_C\ I_{R_2})$, $(VALUE\ (ANGLE\ I_C\ V_C)$ (45 90)), etc. The circuit model constrains the expected behavior, which is to be compared to the observed behavior. This comparison is made following the clustering structure of the circuit, looking for irregularities in the clusters and primitive elements. If an inconsistency is found, the corresponding fault candidate is reported. Note that even if a cluster's behavior is found consistent, we still need to diagnose its components, because there might be a fault in one of its (sub) components that is not reflected at this level. The algorithm is shown in figure 10.

```
cand_fault(cluster, exp_beh, obs_beh)
  if primitive(cluster)
    if inconsistent(cluster, exp_beh, obs_beh)
      report_fault(cluster, exp_beh, obs_beh)
    else
      OK
  else
    if cluster_inconsistent(cluster, exp_beh, obs_beh)
      report_cluster_fault(cluster, exp_beh, obs_beh)
    else
      cand_fault(comp1(cluster), exp_beh, obs_beh)
      cand_fault(comp2(cluster), exp_beh, obs_beh)
```

Figure 10: Algorithm for candidate faults

Table 1 shows an example of the cand_fault procedure, applied to the observed behavior of figure 9. In this example, we start by checking the constraints for cluster S_2; while no contradictions are found at this level, we need to continue verifying the rest of the circuit, traversing its structure. We continue checking S_1 and P_1, until we find that the phase angle of the current and voltage in the capacitor does not correspond to the model of that element. By the characteristics of the observation, we conclude that *"the capacitor is leaking"*. In other words, it is shorted by a small resistance (see figure 11).

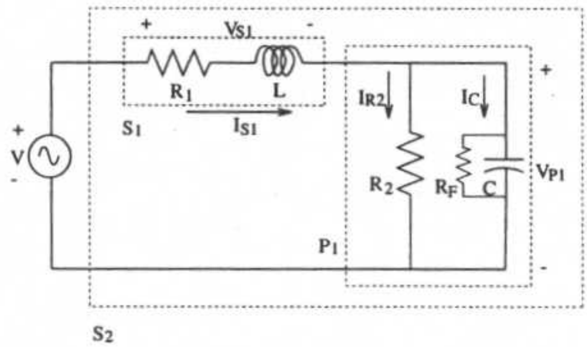

Figure 11: Diagnosed fault

Finding a candidate fault is not enough, we need to make sure that the suggested fault is indeed producing the faulty behavior. This can be done by modifying the circuit according to the suggested fault and performing diagnosis on the modified circuit. If no fault is found, we can tell the suggested fault is a good candidate and report it to the user. Figure 12 shows the algorithm for diagnosis, assuming a single fault exists.

```
diagnosis (circuit, exp_beh, obs_beh)
  fault = cand_fault(circuit, exp_beh, obs_beh)
  faulty_circ = insert_fault(circuit, fault)
  faulty_beh = QPA(faulty_circ)
  if null(cand_fault(faulty_circ, faulty_beh, obs_beh)
    report(fault_cand)
  else
    report(no_solution)
```

Figure 12: Algorithm for diagnosis

Typical faults include open-circuit, short-circuit, and short-circuit with resistance. Under those conditions, the necessary modifications to the original circuit to produce the faulty circuit are simple. Open- or short-circuiting a cluster eliminates that cluster and all its elements, while inserting a fault resistor creates a new cluster. In either case, some variables disappear or new ones appear; to be able to compare the observed behavior with the expected one, we need to have the

Region	Constraint	Observation	Action
S_2	$(VALUE\ I_{S_2}\ 0)$	×	No open circuit
	$(VALUE\ V_{S_2}\ 0)$	×	No short circuit
	$(= I_{S_2}\ I_{S_1})$	√	Currents in cluster OK
	$(= I_{S_2}\ I_{P_1})$	√	
	$(= V_{S_2}\ (+\ V_{S_1}\ V_{P_1}))$	√	Voltages in cluster OK
	$(VALUE$ $(ANGLE\ I_{S_2}\ V_{S_2})\ (0\ 45))$	√	Acceptable phase angle
S_1	...		
P_1	...		
R_2	...		
C	$(VALUE\ I_C\ 0)$	×	No open circuit
	$(VALUE\ V_C\ 0)$	×	No short circuit
	$(VALUE$ $(ANGLE\ I_C\ V_C)\ 90)$	$(VALUE$ $(ANGLE\ I_C\ V_C)\ (45\ 90))$	$\rightarrow (SCR\ C)$ $(connect\ R_F$ $(n1\ C)\ (n2\ C))$ $(<\ Z_{R_F}\ Z_C)$

Table 1: Diagnosis procedure

same set of variables. In the first case, we just set all disappearing variables (i.e. voltages and currents) to zero. In the second case, we rename the new cluster, to have the name of the faulty element, and append the suffix F to the faulty element, so the comparison between the two sets of constraints makes sense (see figure 13).

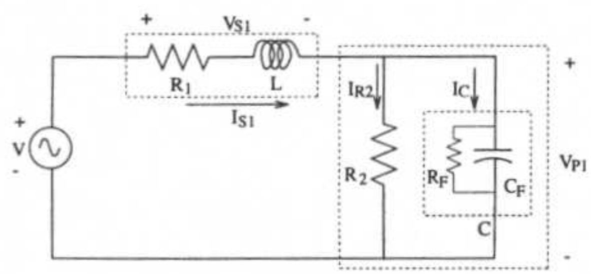

Figure 13: Renaming elements

To handle multiple faults, the diagnosis procedure (figure 12) must be modified to try all possible subsets of the candidate faults. Our approach is similar to that taken by a technician or engineer in a troubleshooting task. The circuit's faulty behavior is observed, the circuit is analyzed, and the expected behavior is obtained. The behaviors are compared and a set of fault candidates is proposed. The technician then tries replacing a (possibly) faulty component, and verifies that the circuit's behavior is now correct. Different subsets of faulty components are tried, until the circuit is fixed.

Order of magnitude reasoning and diagnosis can be combined. Such a combination should allow the system, for example, to prove that a short-circuit with resistance, for which the fault resistance is very small (negligible), is equivalent to a circuit with a perfect short (i.e. with zero resistance).

An Application Domain

Given the basic reasoning abilities this framework offers, one domain of application suitable for using QPA is Power Systems Analysis and Control. Power Systems are modeled by linear circuits (Grainger & Stevenson 1994; Gönen 1988), with lumped, constant parameters, and are normally operated under sinusoidal steady state. These are exactly the kind of circuits QPA reasons about. By using QPA, we can solve important problems in the area of power system analysis. Some of those problems are

Power factor correction. Industrial loads are typically composed of resistive and inductive elements, therefore having a lagging power factor. Connecting a capacitor bank in parallel with the load corrects its power factor. This problem can be solved using a set of rules to propose a solution (i.e. a modified power system).

Power distribution. Another problem that requires structural changes is that of distribution of power transmission between parallel transmission lines. This problem arises in situations where the transmitted power increases, and one of the lines is not capable of holding the resulting amount of current. In that case, the problem is solved by rerouting part of the current through transmission lines that still have some capacity. That can be accomplished by installing capacitors, tap changing or phase shifting transformers in series with the transmission lines.

These problems are solved by designing structural changes to the power system. The resulting power system can be modeled as a linear circuit and analyzed using QPA to verify that the design goal has been accomplished.

As an example, consider two parallel transmission lines with equal inductive reactance, as shown in Figure 14. The currents through both lines are equal

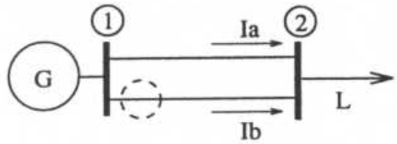

Figure 14: Power Distribution Problem

and we want to design a solution that ensures that ($<$ I_a I_b). The dashed circle indicates where the correction element should be placed.

The simplest method to redistribute the current is by insertion of a capacitor in the place indicated by the dotted circle of figure 14.

The resulting power system can be modeled by the circuit shown in figure 15.

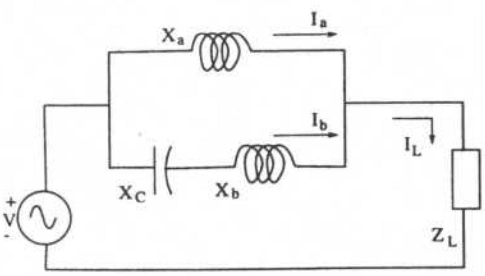

Figure 15: Rerouting current by inserting a capacitor bank

Analyzing that circuit using QPA, we can verify that indeed ($<$ I_a I_b), as stated in the problem (see figure 16).

Notice that reasoning about the circuits in terms of phase angles is crucial to the solution of these problems. This kind of reasoning has been made possible by extending the circuit ontology to include phase angles and phasor elements. This capability is unique of QPA. Previous work in the field would be unable to solve these kinds of problems as they did not include this element in the representation.

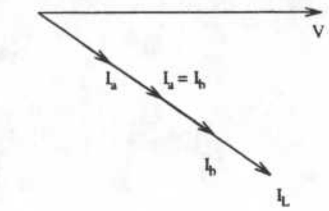

Figure 16: Solution to the Power Distribution Problem

Furthermore, we hope to apply QPA and PSAD to electrical engineering education, training of power systems operators, etc. Simulation programs only yield numerical results, giving the student no information about why results appear in the solution of a problem, or how a given solution was found. It will be very useful for a student to get a chain of causal effects to questions like "What happens to V_5 if there is a short-circuit between nodes 3 and reference?", "What happens to V_3 when V_5 increases?", or "Suppose transformer $T1$ changes to a higher tap, how does the flow of power in line from buses 5 to 3 change?". Such questions appear throughout books on Power System Analysis, see for example (Grainger & Stevenson 1994, problem 3.13 on page 139; problem 7.16 on page 282; problem 9.17 on page 379).

Implementation and Evaluation

The system is being implemented in Allegro Common Lisp for Sun Workstations. The system consists of three layers: Power System Analysis and Design (PSAD), Qualitative Phasor Analysis (QPA), and Multiple Set Constraint Propagation (MSCP).

The current interface designed for this project is a textual symbolic description of the input and output, plus a graphical rendition of a phasor diagram representative of a set of behaviors. The input is the topological configuration of the circuit or power system, which includes the definition of each element and their interconnections. The input constraints are in the form mentioned in the preceding subsection. The output of the system is a set of constraints, representing the partially constrained model of the circuit or power system. In the case of diagnosis or control design, the new topological configuration of the circuit will be returned to the user.

The system will be evaluated by comparing its results with examples found in textbooks and by analytical tools used in circuit analysis. In the case of analysis, for a given circuit, we can assign numerical values to its parameters and run a numerical simulation; from those values extract qualitative properties of the circuit (constraints); feed the circuit and constraints to QPA and compare the resulting circuit model and phasor diagram with the numerical results of the simulation. In the case of diagnosis, a circuit can be given to QPA to be analyzed. We can insert arbitrary faults into that circuit, simulate the faulty circuit numerically and extract its qualitative properties (expressed as constraints). Finally we compare the simulated fault with the diagnosed fault.

Related Work

Several systems have been built to reason about and derive the behavior of electrical circuits. Most of them focus on either digital circuits or DC analog circuits (see for example, DeKleer (de Kleer 1984), Hamscher (Hamscher 1991), Williams (Williams 1984)). (Sussman & Steele 1980) mention the possibility of performing analysis of linear electrical circuits in sinusoidal steady state by the use of constraint. In their paper *Constraints*, they perform all the analysis and derive the set of constraints; their program takes the constraints and uses them to design (i.e. compute values of) the different parameters of the circuit. QPA, on the other hand, derives the circuit model automatically and uses knowledge of electric circuit theory to perform analysis, elementary diagnosis, and design.

The solution of electrical circuits by differential equations is adequate if we are analyzing its behavior in transient state, but not for its solutions in steady state. QSIM (Kuipers 1985) can simulate the behavior of linear circuits, but since it is based on differential equations, its scope is limited to transient state analysis. That formalism is not able to represent a sinusoidal source in terms of allowed set of constraints. The response description normally given by QSIM is at a microscopic level with respect to time, describing the possibilities at each distinguished time point. It is a well known fact that all variables in a circuit in steady state will be steady sinusoidals; there is no point in trying to find out if a peak (defined by a landmark) will be greater, equal or less than the next one. That microscopic view prevents us from getting the big picture of what is happening in the circuit and only gives place to ambiguity.

DeKleer's confluences (de Kleer & Brown 1984) allows us to reason about change, but only in terms of magnitudes of *scalar* quantities. Since the main tool used to solve this steady state problem is *phasors* (e.g. a particular kind of vector), we need a way to represent angular information and the interaction between the magnitudes of different quantities and their phase angles.

Trying to describe an electrical circuit in terms of processes is awkward. Similar to Kuiper's approach, the kind of description that QPT (Forbus 1984) yields would be at the microscopic level. This kind of representation would be probably talking about charges, and how the process of moving charges (an electrical current) would result from the application of an electric field. We need something at a higher level of abstraction, where the stable oscillation of alternating currents is a well known model and not the goal to be established. Forbus' approach is not suitable to directly solving the problem of analysis of electrical circuits in sinusoidal steady state.

Among the few that have worked with power systems, Struss (Struss 1992a; 1992b) has developed a system for diagnosing faults in power transmission networks. He uses a relational approach to model power system components and consistency-based diagnosis to find faults in the system, based on the reading of "distance protection relays". A component's behavior is described in terms of local variables, captured in a relation where each tuple defines a possible mode of operation. In other words, the diagnosis is based on what elements a breaker is protecting, what breakers tripped, and what breakers did not in a given situation, the distance from breakers to faults, etc. In the presence of a fault, the observed behavior will produce inconsistencies with the expected behavior, and those inconsistencies will suggest a set of candidate fault sets. Each fault set is then compared against observations, to verify constrain consistency. This representation is not based on circuit theory (i.e. Kirchoff laws), and does not accounts for the behavior of the system at the level of electrical circuits. QPA is focused on the understanding of behavior of electrical circuits, performing diagnosis based on its components' behavior and on phase angle information.

Another important characteristic of QPA is its ability to simplify a circuit based on order of magnitude relations. If based on order of magnitude relations given by the user, and appropriately propagated, QPA determines that a current (voltage) is near zero, it can discard that part of the circuit, replacing it by an open (short) circuit. We call this feature *structural exaggeration.* A similar kind of transformation is presented by Liu's ARC (Liu 1991). Based on different operating regions of components, parts of the circuit can be eliminated (mainly due to a component acting as an open circuit). The system is then recast, based on its new topological configuration. As mentioned above, Struss presents a diagnosis system that works with models at different levels of abstractions. The simplifications presented in that work deal with the internal model of each device, refining it to yield more accurate results when necessary. The overall structural description of the system does not change with the use of different models. Structural exaggeration, in contrast, simplifies the overall structure of the circuit, based on existing behavior constraints.

Conclusion

As pointed out in the preceding sections, we have developed a representation that enables us to reason about linear circuits in sinusoidal steady state. This representation is a qualitative version of one of the main tools used in electrical engineering, *phasor analysis*. The main idea is to represent the circuit by a set of constraints that limits the set of allowed behaviors of the circuit. Based on this representation, we can perform qualitative analysis of electrical circuits, covering both zeroth- and first-order reasoning.

There are two main fields of applications for this work that we explore. Power system analysis and control design was introduced above. That section mentions two main problems that will be solved by using QPA, power factor correction and power distribution on transmission lines. Another potential application is the use of QPA in education. QPA is an analytical tool that not only returns numerical results from a simulation of a circuit, but is also able to reason about the circuit in the same terms found in the explanations given in text books. That constitutes an important tool for the student of electrical circuits to really understand what is happening inside the circuit, what would happen if parts of the circuit change or if the operating conditions change.

The main contribution of QPA is that, by extending the circuit ontology to include phasors and by using a constraint-based model of the circuit, we can solve a wider range of problems in the field of qualitative reasoning about complex linear systems. In developing the project, we will address problems such as: what modifications need to be done to normal constraint propagation procedures to deal with constraints of different kinds?; How can the structure of a circuit be simplified, based on order of magnitude information derived from constraint propagation?; Can we design solutions for the problems of operation, diagnosis, and control of power transmission systems, based on first principles of phasor analysis?

To demonstrate the expressive power of QPA, we have worked out some examples that show it has the inferential power we need to successfully perform the reasoning task we have in mind. This conclusion has been supported with the implementation we have so far and will be further explored throughout this project.

References

Boyce, W. E., and DiPrima, R. C. 1969. *Elementary Differential Equations*. New York: John Wiley, second edition.

de Kleer, J., and Brown, J. S. 1984. Qualitative physics based on confluences. *Artificial Intelligence* 24:7–83. Also in *Readings in Knowledge Representation*, Brachman and Levesque, editors, Morgan Kaufmann, 1985, pages 88-126.

de Kleer, J. 1984. How circuits work. *Artificial Intelligence* 24:205–280.

Fitzgerald, A. E. 1945. *Basic Electrical Engineering*. New York and London: McGraw-Hill.

Forbus, K. D. 1984. Qualitative process theory. *Artificial Intelligence* 24:85–168.

Gönen, T. 1988. *Modern Power System Analysis*. New York: John Wiley and Sons.

Grainger, J. J., and Stevenson, W. D. 1994. *Power System Analysis*. New York: McGraw-Hill.

Hamscher, W. 1991. Modeling digital circuits for troubleshooting. *Artificial Intelligence* 51:223–271.

Hyvönen, E. 1989. Constraint reasoning based on interval arithmetic. In *Proc. 11th Int. Joint Conf. on Artificial Intelligence (IJCAI-89)*, 1193–1198.

Kerr, R. B. 1977. *Electrical Network Science*. Englewood Cliffs, NJ: Prentice-Hall.

Kuipers, B. J. 1985. The limits of qualitative simulation. In *Proc. 9th Int. Joint Conf. on Artificial Intelligence (IJCAI-85)*, 128–136. San Mateo, CA: Morgan Kaufmann.

Liu, Z.-Y. 1991. Qualitative reasoning about physical systems with multiple perspectives. Technical Report CIS-TR-91-04, University of Oregon.

Struss, P. 1992a. An application of model simplification and abstraction to fault localization in power transmission networks. In *Working notes of the workshop Approximaion and Abstraction of Computational Theories*, 205–212.

Struss, P. 1992b. A theory of simplification and abstraction for relational models. In *Working notes of the workshop Approximaion and Abstraction of Computational Theories*, 213–226.

Sussman, G. J., and Steele, G. L. 1980. CONSTRAINTS: a language for expressing almost-hierarchical descriptions. *Artificial Intelligence* 14:1–39.

Williams, B. C. 1984. Qualitative analysis of MOS circuits. *Artificial Intelligence* 24:281–346.