CHAPTER 11

NUMERICAL METHODS FOR TRANSPORT-RESISTANCE SOURCE–SINK ALLOCATION MODELS

P. PRUSINKIEWICZ[#], M. ALLEN[##],
A. ESCOBAR-GUTIÉRREZ[###], AND T.M. DEJONG[##]

[#]Department of Computer Science, University of Calgary,
Calgary, Alberta, Canada
[##]Department of Plant Sciences, University of California, Davis, California, USA
[###]Unité d'Ecophysiologie des Plantes Fourragères, INRA, Lusignan, France

Abstract. A key problem of functional-structural plant modelling is the partitioning of resources between plant components. One approach to addressing this problem is based on the simulation of transport between the sources and sinks of a resource. To this end, we exploit an analogy between the flow of resources in a plant and the flow of currents in an electric circuit, and we present a computationally effective method for solving this circuit. The method lends itself to a straightforward implementation using L-systems, and is thus well suited for simulating the partitioning of resources in spatially-explicit models of growing plants.

INTRODUCTION

We present a method for simulating the acquisition, transport and partitioning of a resource within a plant. This method operates at the level of plant architecture, with a plant being viewed as a branching arrangement of metamers or phytomers (Bell and Bryan 1991; Room et al. 1994). A metamer consists of an internode with the associated lateral organs: buds, leaves, flowers or fruits (Figure 1a). We conceptualize each metamer as a conduit element with a single source or sink attached laterally at the distal end of the conduit element (Figure 1b). More complex metamers, which may include several lateral organs, or store resources in the internodes, can be modelled by combining individual sources or sinks into one resultant component. The simulation method exploits an analogy between pressure-driven fluid flow in plant vasculature and current flow in an electric circuit (Figure 1c). This circuit may be *nonlinear*, which means that parameters of the components

J. Vos, L.F.M. Marcelis, P.H.B. de Visser, P.C. Struik and J.B. Evers (eds.), Functional-Structural Plant Modelling in Crop Production, 123-137.

depend on the potentials and currents in the circuit. It may also be *non-stationary*, with parameters depending on time either explicitly or as a result of the accumulation of resources in plant organs. Finally, the circuit may have a *dynamic structure*, which means that its configuration changes over time, consistent with the addition or loss of metamers in the course of plant development.

A circuit, such as that shown in Figure 1c, represents a snapshot of the modelled plant at a particular point in time. During the simulation, this snapshot is used to calculate flow rates and the amounts of the resource transported from sources to sinks over a small time increment. The circuit is then updated to reflect the resulting changes, and the next iteration of the simulation proceeds.

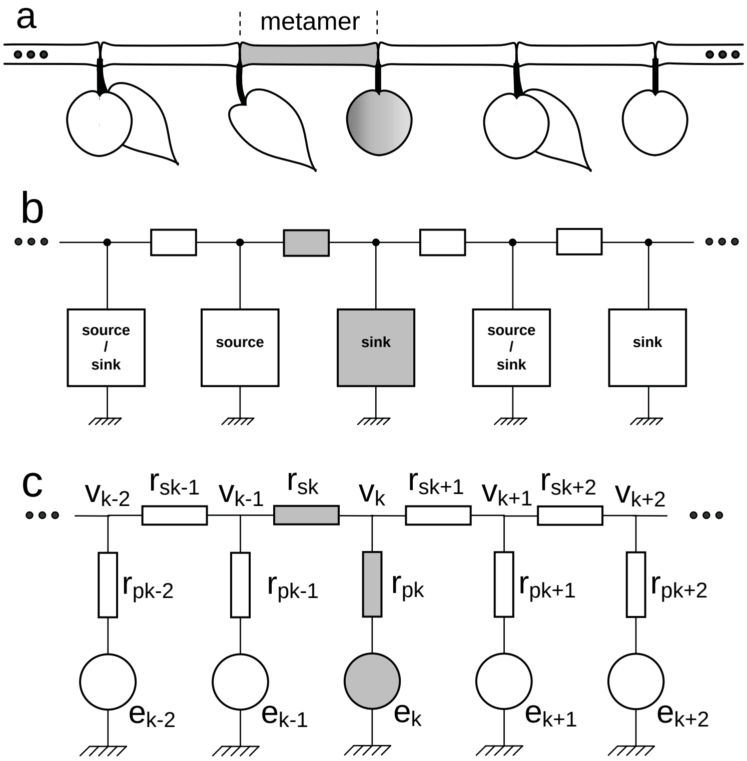

Figure 1*. A branch axis (a) is abstracted into an assembly of conduits, sources and sinks (b), and a corresponding electric circuit (c). The axis is a sequence of metamers, each of which consists of an internode and optional lateral organs, such as leaves and fruits, attached at the distal end of the internode. Metamer k is shown in grey. Symbols:* r_{sk} *– resistance of the conduit;* e_k *– electromotive force of the source or sink;* r_{pk} *– internal resistance of the source or sink,* v_k *– node potential associated with metamer k*

THE ANALOGY BETWEEN A PLANT AND AN ELECTRIC CIRCUIT

Our presentation is expressed in terms of carbon partitioning, although a similar approach may apply to other resources, such as water. According to the Münch hypothesis, carbon flow in the phloem is driven by differences in hydrostatic pressure (hydrostatic potential) between sources and sinks (Salisbury and Ross 1992; Nobel 2005). This leads to the following simplified formula for the volume flow rate I_k of the phloem sap (Thornley and Johnson 1990; Minchin et al. 1993; Bidel et al. 2000; Daudet et al. 2002):

$$I_k = \frac{P_{k-1} - P_k}{R_{Hk}} . \tag{1}$$

The numerator $P_{k-1} - P_k$ represents the pressure difference between the proximal and distal node of internode k, and the denominator R_{Hk} is the hydraulic resistance of phloem in internode k to the movement of the solution. This resistance is determined by Poiseuille's law (Nobel 2005),

$$R_{Hk} = \frac{8 l_k \eta}{\pi n_k \rho_k^4} , \tag{2}$$

where l_k is the internode length, η is the solution viscosity, n_k is the number of sieve tubes in the internode, and ρ_k is the radius of the sieve tubes (the tubes are assumed to have circular cross-sections and the same radius).

Equation (1) is formally equivalent to Ohm's law,

$$i_k = \frac{v_{k-1} - v_k}{r_k} , \tag{3}$$

which states that the electric current i_k flowing through a resistive element is proportional to the difference of the potentials $v_{k-1} - v_k$ at the element's terminals, and inversely proportional to the element's resistance r_k. Furthermore, the mass of the solution flowing into any node is equal to the mass flowing out of it (the mass is conserved), and the sum of pressure differences measured along any closed loop is equal to zero. These are hydraulic analogues of Kirchhoff's laws for electricity (Ramakalyan 2005). The analogies between the laws governing fluid flow and Ohm's and Kirchhoff's laws make it possible to describe the transport of carbohydrates in a plant in terms of electric circuits, for which a wealth of useful results, such as rules for transforming circuits to an equivalent form, is readily available. The correspondence between hydraulic and electric entities is summarized in Table 1.

Table 1. *Analogies between hydraulic/physiology and electric entities*

Physiological/hydraulic entity	Electric entity	Symbol
mass or volume	charge	q
mass or volume flow rate [a)]	current	i
hydrostatic potential, pressure	electric potential	v
pressure difference	potential difference, voltage [b)]	v, e
hydraulic resistance	resistance	r
hydraulic conductance	conductance	g

[a)] Flow rate is the mass or volume that flows past a given cross-sectional area per unit time.

[b)] We also use the term electromotive force (EMF) to characterize voltage sources.

SOLVING LADDER AND BRANCHING LADDER CIRCUITS

The circuit representing a single branch (Figure 1c) is an example of a *ladder circuit*, i.e. a circuit resulting from a sequence of repetitive connections of electric components in series and in parallel (Ramakalyan 2005). Let us first assume that these components are linear, meaning that the resistances and electromotive forces do not depend on potentials and currents. We can then solve the circuit, i.e. find all potentials and currents, in a manner that only requires a local information transfer between adjacent segments of the ladder (adjacent metamers). Furthermore, only unilateral information flow, either in the basipetal or in the acropetal direction, is needed in each phase of computation. This type of information transfer is well supported by L-systems with fast information transfer (Karwowski and Prusinkiewicz 2003; Prusinkiewicz et al. this volume, Chapter 3), which therefore provide a convenient framework for implementing the solution to the ladder circuits as a part of functional-structural plant models (see section "*Implementation*").

The solution makes use of the rules for series or parallel connection of linear electric circuits (Figure 2). These rules represent a special case of Thévenin's theorem, which states that it is possible to simplify any linear circuit, no matter how complex, to an equivalent circuit consisting of a single source of electromotive force and a series resistance (Ramakalyan 2005).

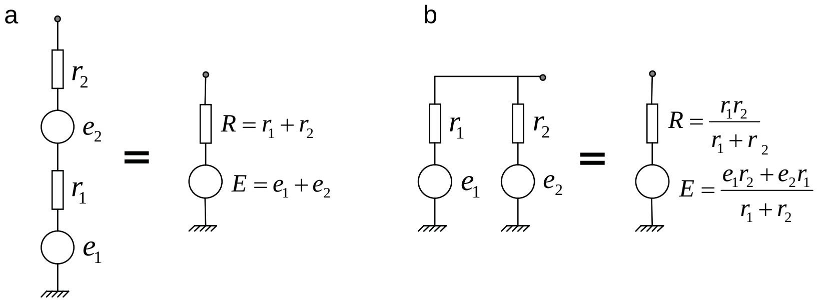

Figure 2. *Rules for finding circuits equivalent to a series (a) and parallel (b) connection of linear components*

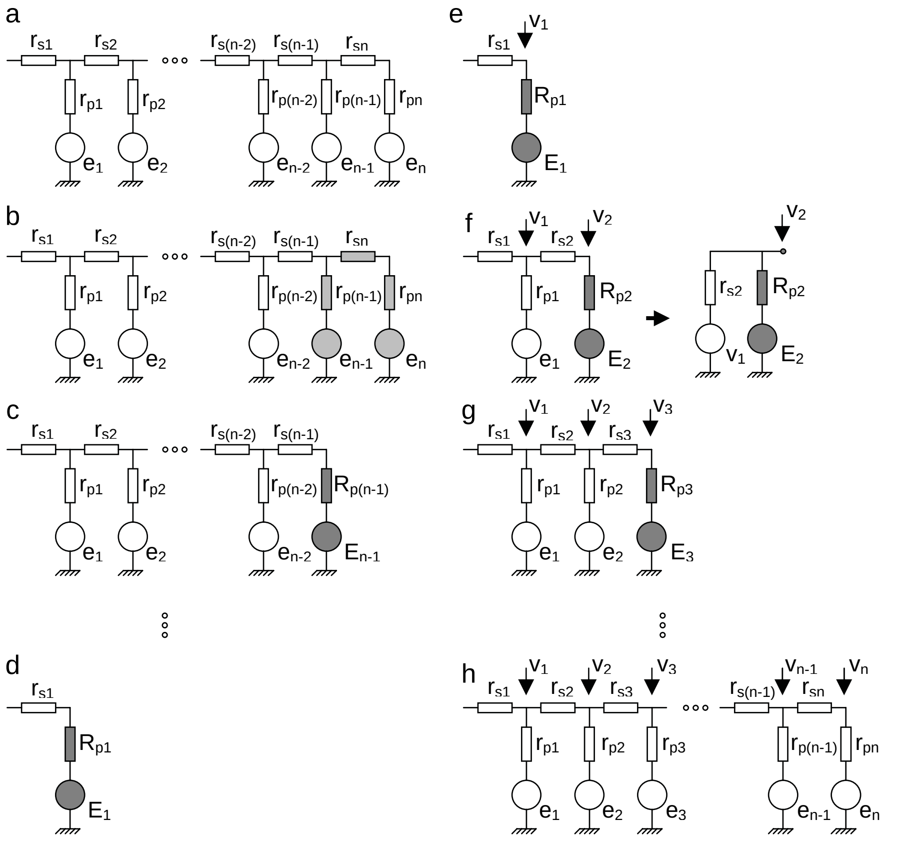

***Figure 3**. Solving a ladder circuit by a sequence of folding (a-d) and unfolding (e-h) operations. The dots between figures c-d and g-h indicate a continuation of the folding and unfolding processes. Down arrows labelled v_1, v_2,...,v_n indicate node potentials, computed one after another as the circuit is unfolded*

The ladder circuit is solved in two phases, which we term *folding* and *unfolding* of the circuit. During the folding phase (Figure 3a-d), pairs of distal (rightmost) elements of the ladder are recursively reduced to simpler elements using the rules of Figure 2. This reduction proceeds until the simple circuit shown in Figure 3d is found. Its node potential v_1 is equal to E_1, since there is no current flowing in that circuit (the resistance r_{s1} is formally present only to assure a uniform representation of all metamers, which simplifies programming), and thus there is no voltage drop on the resistance R_{p1} (Figure 3e). To find potential v_2 (Figure 3f), we formally treat potential v_1 as an electromotive force. Referring to Figure 2b, we then calculate v_2 as

$$v_2 = \frac{v_1 R_{p2} + E_2 r_{s2}}{r_{s2} + R_{p2}}. \tag{4}$$

Given potential v_2, we unfold the next element of the ladder and calculate potential v_3 (Figure 3g). By continuing this process recursively, we find all the remaining node potentials (Figure 3h). With the potentials known, all currents can be calculated using Ohm's law.

An extension of this method to a branching ladder circuit is illustrated in Figure 4. The terminal axes are folded separately up to the branching point **B** (Figure 4a,b). The circuits r_{s1} R_{p1} E_1 and r_{s2} R_{p2} E_2 that comprehensively represent these axes are then combined into a single equivalent circuit R_{s12} R_{p12} E_{12} (Figure 4c) using the transformation rule of Figure 2b. For technical reasons related to the L-system implementation, this operation is combined with a folding step, resulting in the circuit shown in Figure 4d. The resulting ladder is folded and unfolded as in Figure 3 to yield the potential v of the branching point (Figure 4e). At this stage, the circuit R_{s12} R_{p12} E_{12} is reverted to the parallel connection of its components r_{s1} R_{p1} E_1 and r_{s2} R_{p2} E_2, so that the node potentials v_1 and v_2 of the first metamer in each axis can be found (Figure 4f). The remaining potentials and currents are then calculated by unfolding each axis independently (Figure 4g). Circuits with many branching points are solved in a similar way, by repetitively merging axes into equivalent circuits while folding, and reverting these circuits to parallel connections of axes while unfolding.

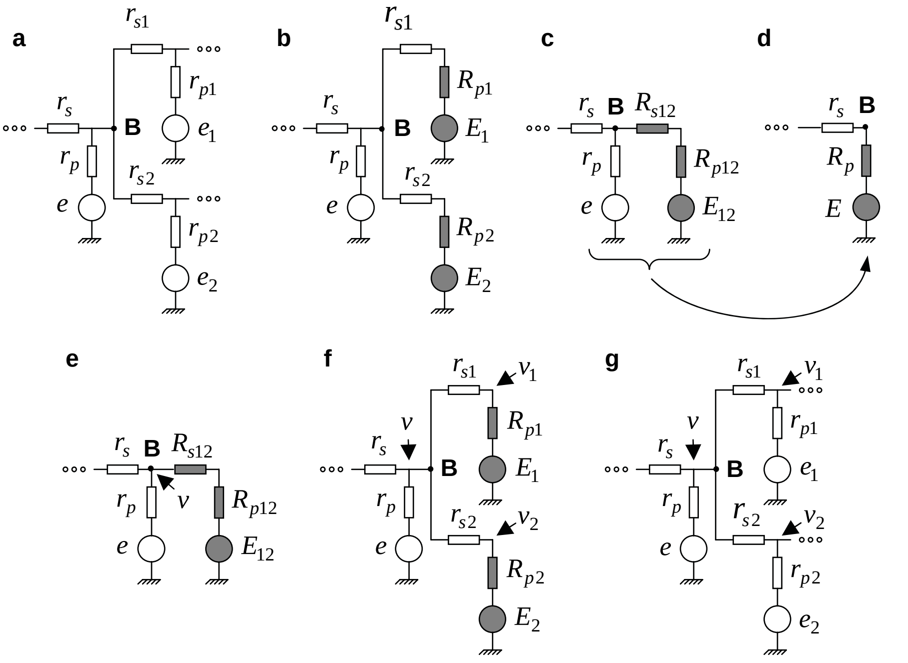

Figure 4. *Generalization of the folding–unfolding process to a ramified circuit*

SOLVING NONLINEAR CIRCUITS

The method presented above only applies to linear circuits, but circuits that model sink–source relations in a plant are likely to be nonlinear. We solve nonlinear circuits using an iterative method, in which the solution is sought through a series of linear approximations of circuit components. Mathematically, this approach amounts to numerically solving a system of nonlinear equations using the Newton-Raphson method (Press et al. 1992).

Let us consider the simple linear circuit shown in Figure 5a. To solve it, we may first divide it into two parts, as shown in Figure 5b. The dependence of currents i_1 and i_2 on the node potentials v_1, v_2 is then expressed by the equations:

$$i_1 = \frac{v_1 - e_1}{r_1 + r_s} \quad \text{and} \quad i_2 = \frac{v_2 - e_2}{r_2} \,. \tag{5}$$

The behaviour of each part is represented graphically by a straight line in Figure 5c. If we now reconnect both parts to return to the circuit of Figure 5a, the node potentials will be the same, $v_1 = v_2 = v$, while the currents flowing into each branch will add up to 0, yielding $-i_1 = i_2 = i$. The solution to the circuit of Figure 5a will thus be represented by the point in which the lines characterizing both parts intersect.

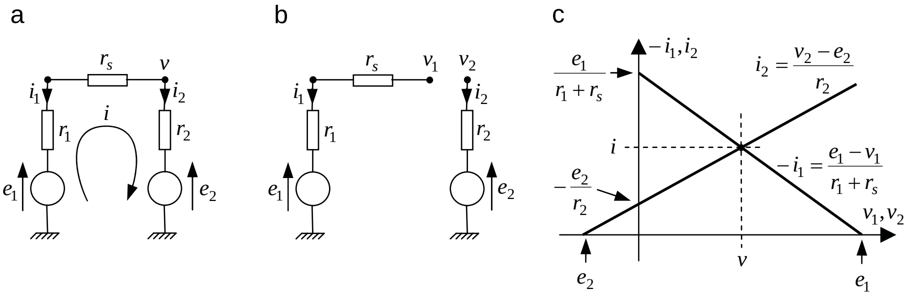

***Figure 5**. Graphical solution of a simple circuit. The circuit (a) is divided into two parts (b), represented by straight lines in the voltage-current plane (c). The solution to the circuit (a) corresponds to the intersection point (i, v) of these lines*

The above method of solving a circuit by intersecting voltage-current characteristics carries over to nonlinear circuits. For example, Figure 6a depicts the case in which the source (e_1, r_1) and the conduit r_s are linear, but the sink (e_2, r_2) is not. We find the solution iteratively, beginning with an initial estimate $v^{(1)}$ of potential v. First, we approximate the sink by the best-fitting linear circuit at the potential value $v^{(1)}$. This linearized sink circuit is represented by the tangent to the function $i_2(v)$ at the point ($v^{(1)}$, $i_2(v^{(1)})$) (Figure 6b). Its conductance and electromotive force can thus be expressed as:

$$g_2 = \frac{1}{r_2} = \left.\frac{di_2(v)}{dv}\right|_{v^{(1)}} \quad \text{and} \quad e_2 = v^{(1)} - r_2 i_2\left(v^{(1)}\right). \tag{6}$$

The circuit consisting of the source, conduit and linearized sink is then solved as in Figure 5, which leads to a new estimate $v^{(2)}$ of the node potential v, and a corresponding estimate $i'(v^{(2)})$ of the current in the circuit (Figure 6c). A comparison of this latter estimate with the actual current $i_2(v^{(2)})$ that would flow through the nonlinear sink given node potential $v^{(2)}$ shows a significant error (Figure 6c). The sink is thus linearized again, this time at the point ($v^{(2)}$, $i_2(v^{(2)})$) (Figure 6d), which leads to the next estimate v_3 of the node potential v (Figure 6e). After one more iteration step, a solution with an acceptably small error is found, ending the iteration (Figure 6f).

One advantage of the Newton-Raphson method is that it extends to systems with many nonlinear components (Press et al. 1992). On the other hand, like many other iterative methods, it is not guaranteed to converge. In the practice of functional-structural modelling (Allen et al. 2005), we occasionally experienced convergence problems, especially when adding new metamers to a developing structure. These problems could be alleviated by properly choosing the initial values of resistances in the new metamers (small conduit resistance r_s and large source/sink resistance r_p), then adjusting them gradually to the desired values to avoid sudden jumps in the solution.

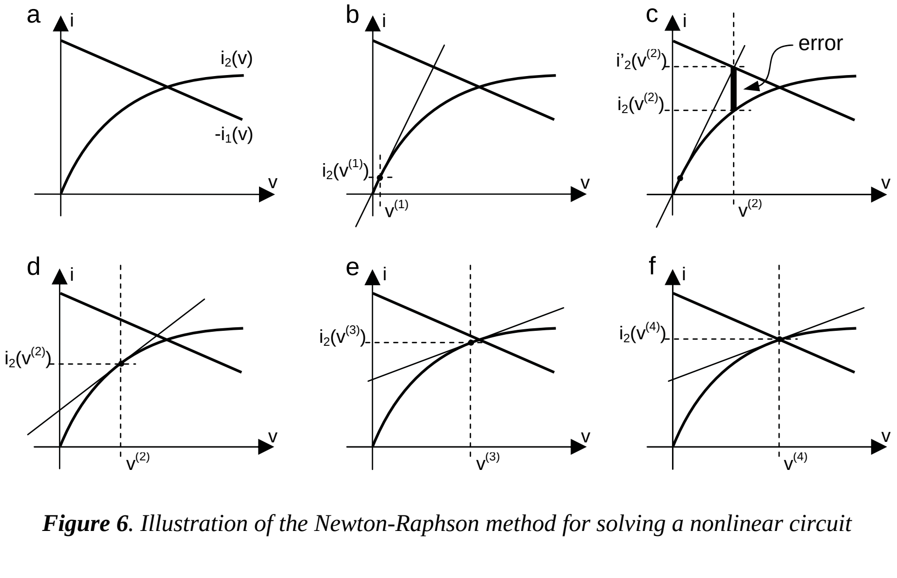

Figure 6. *Illustration of the Newton-Raphson method for solving a nonlinear circuit*

IMPLEMENTATION

The described method for solving a circuit can be incorporated into functional-structural plant models by organizing the simulation into two iteration loops (Figure

7). This is similar to the organization of the biomechanical simulation presented by Prusinkiewicz et al. (this volume, Chapter 3). In the inner loop, potentials and currents at a given simulation time are calculated iteratively, by linearizing the circuit and solving it using the folding and unfolding operations. The iteration proceeds until the cumulative error stemming from the linearization of nonlinear circuit components is sufficiently small. Once the currents are found, charges are transported between sources and sinks in a time-advancing forward integration step within the outer iteration loop. Additional charges and changes to the circuit structure may also be introduced in this step, for instance to simulate photosynthesis or to create new metamers at branch apices. This leads to a modified circuit, in which the values of component parameters reflect the new charges and the structure reflects the possible addition or loss of metamers. The updated circuit is then ready for the next iteration cycle within the inner loop.

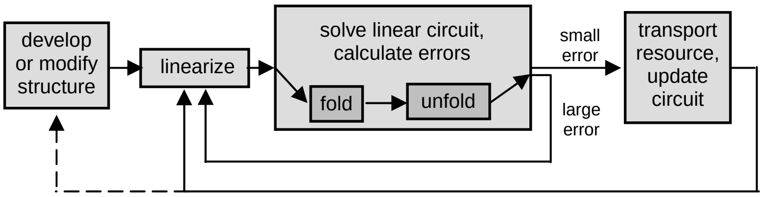
develop or modify structure
linearize
solve linear circuit, calculate errors
fold
unfold
small error
large error
transport resource, update circuit

***Figure 7**. Phases of simulation. Each phase corresponds to an L-system derivation step. The L-system string is scanned from right to left in the folding phase, and from left to right in the remaining phases*

The computations can be conveniently specified in the L-system-based L+C modelling language (Karwowski and Prusinkiewicz 2003; Prusinkiewicz et al. this volume, Chapter 3). A key data structure describes a linear circuit:

```
struct LinCircuit
{
   float e; // EMF
   float r; // resistance
};
```

The function `in_parallel()` implements the parallel connection of two linear circuits according to Figure 2b:

```
LinCircuit in_parallel(LinCircuit c1,LinCircuit c2)
{
  LinCircuit c;
  c.e = (c1.e * c2.r + c2.e * c1.r) / (c1.r + c2.r);
  c.r = (c1.r * c2.r) / (c1.r + c2.r);
  return c;
}
```

The `LinCircuit` data structure is used as the basis for defining a metamer:

```
struct MetamerData
{
  int type;       // e.g. leaf, fruit, axis start, end
  float rs;       // conduit/internode resistance
  LinCircuit cp;  // source/sink EMF ep, resistance rp
  LinCircuit ceq; // circuit resulting from folding
  float v;        // node potential
  float i;        // current through source/sink
  float q;        // accumulated charge
  /* ... other characteristics, e.g. age, size */
  void update_charge(); // account for flow, photosynth.
  void linearize_circuit(); // implements Eq. 6.
};

module M(MetamerData);
```

The `update_charge()` function in the `MetamerData` structure is called in each time-advancing simulation phase to increment or decrement the charge `q` accumulated in the metamer. This increment or decrement is proportional to the current `i` and, in the case of leaves, the rate of photosynthesis. The updated charge is a parameter to a function that determines the nonlinear potential-current characteristic of the source or sink. Some methods for specifying such functions are presented by Allen et al. (2005). Given that characteristic, the `linearize_circuit()` function is called to calculate the electromotive force and resistance of the linear circuit `cp` that approximates the sink or source at node potential `v`. Once these values have been determined for all metamers, the resulting ladder circuit is ready to be solved by the folding and unfolding operations.

Folding of an axis begins with the production

```
M(md) :
{
  if (md.type == end)
  {
    md.ceq = md.cp;
    produce M(md);
  }
}
```

which formally defines the result `ceq` of folding the most distal metamer of this axis. The folding then proceeds by scanning the L-system string from right to left, and applying the following production to the remaining metamers:

```
M(md) >> M(mdr) :
{
  mdr.ceq.r += mdr.rs;
  md.ceq = in_parallel(md.cp, mdr.ceq);
  produce M(md);
}
```

This production implements the operations shown in Figure 3 a-d. Branching points are handled separately, according to Figure 4 b-d. Modules `SB` (start branch) and `EB` (end branch) enclose a lateral branch.

```
M(md) >> SB() M(mdr2) EB () M(mdr1) :
{
  mdr1.ceq.r += mdr1.rs;
  mdr2.ceq.r += mdr2.rs;
  LinCircuit mdr = in_parallel(mdr1.ceq, mdr2.ceq);
  md.ceq = in_parallel(md.cp, mdr);
  produce M(md);
}
```

In the unfolding phase, the node potentials of all metamers are found by scanning the string from left to right. Unfolding is initiated by the production

```
M(md) :
{
  if (md.type == start)
  {
    md.v = md.ceq.e;
    produce M(md);
  }
}
```

which brings computation to the state illustrated in Figure 3e. Computation proceeds according to Figure 3f-h, through the repetitive application of production

```
M(mdl) << M(md) :
{
  LinCircuit ml = {mdl.v, md.rs};
  md.v = in_parallel(ml,md.ceq).e;
  produce M(md);
}
```

Since the left context in L-systems refers to the proximal neighbour irrespective of the possible ramifications of the modelled structure, no special production is needed to handle branching points during unfolding.

Given the node potentials `v`, all currents in the linearized circuit can be found using Ohm's law and compared with the values determined by the characteristics of nonlinear sources/sinks for the same values of node potentials. The sum of the absolute values of the differences defines the cumulative error due to the linearization. If this error exceeds a predefined limit, the circuit is relinearized at the new node potentials, and a new estimate of the solution is found in the next round of iteration. Once the cumulative error becomes sufficiently small, the `update_charge()` function is called for each metamer, initiating the next cycle of the simulation.

The above outline of the L-system implementation of the resistance-transport-driven resource allocation model shows that the calculations can be specified in a compact manner, directly reflecting the essence of the method. Compared to implementations using an external equation solver, the integration of numerical calculations into an L-system-based model offers the benefit of automatically updating the system of equations as the simulated structure develops. This is an important advantage when constructing functional-structural models of growing plants.

CONCLUSIONS

The presented algorithm and its L-system implementation offer a practical method for simulating resource flow and partitioning in fixed and growing branching plant structures. The method exploits an analogy between the flow of resources through plant vasculature and current flow in an electric circuit. For linear circuits, computation of potentials and flow rates proceeds in two phases. During the folding phase, the circuit is reduced to a very simple one, for which the node potential can easily be found. During the unfolding phase, node potentials of all metamers are computed in a sequence, from proximal to distal, with each subsequent value being determined on the basis of the previous one. This organization of computation is well suited for implementation using L-systems, because it only relies on information transfer between neighbouring metamers. Furthermore, in both the folding and unfolding phases, the information flows in one direction, either acropetally or basipetally, which leads to a particularly effective implementation using L-systems with fast information transfer. Formally, the presented method is related to the method for solving tridiagonal systems of linear equations using L-systems (Federl and Prusinkiewicz 2004), which in turn is based on Gaussian elimination (Press et al. 1992). We note, however, that our method extends to branching structures for which the equation systems are no longer tridiagonal, yet Gaussian elimination can still be performed (Parter 1961). Nonlinear circuits are solved numerically with the Newton-Raphson method, using a sequence of linear approximations to the given circuit. An application example is given by L-PEACH, a generic functional-structural model of tree development driven by carbon partitioning (Allen et al. 2005; Allen et al. this volume, Chapter 12).

Several problems remain open for further research. They include:

- a refinement of the presented model of carbon allocation, in which the assumption of constant concentration of carbohydrates in the phloem sap would be removed (cf. Smith et al. 1980; Thompson and Holbrook 2003);
- extensions and applications of the model to the transport and partitioning of other resources, such as water and nitrogenous compounds; and
- development of functional-structural models that would incorporate transport and partitioning of several resources.

ACKNOWLEDGEMENTS

We thank Jim Hanan, Alla Seleznyova, Jan Vos, Leo Marcelis, Lynn Mercer, Brendan Lane and the referees for insightful comments. The financial support by the Natural Sciences and Engineering Research Council of Canada Discovery Grant to

P. Prusinkiewicz, a Region Poitou-Charentes grant to A. Escobar-Gutiérrez, and a University of California Davis Department of Pomology / Plant Sciences Faculty Research Grant to T.M. DeJong is gratefully acknowledged.

REFERENCES

Allen, M.T., Prusinkiewicz, P. and DeJong, T.M., 2005. Using L-systems for modeling source-sink interactions, architecture and physiology of growing trees: the L-PEACH model. *New Phytologist*, 166 (3), 869-880.

Bell, A.D. and Bryan, A., 1991. *Plant form: an illustrated guide to flowering plant morphology*. Oxford University Press, Oxford.

Bidel, L.P.R., Pagès, L., Riviere, L.M., et al., 2000. MassFlowDyn I: a carbon transport and partitioning model for root system architecture. *Annals of Botany,* 85 (6), 869-886.

Daudet, F.A., Lacointe, A., Gaudillère, J.P., et al., 2002. Generalized Münch coupling between sugar and water fluxes for modelling carbon allocation as affected by water status. *Journal of Theoretical Biology,* 214 (3), 481-498.

Federl, P. and Prusinkiewicz, P., 2004. Solving differential equations in developmental models of multicellular structures expressed using L−systems. *In:* Bubak, M., Van Albada, G., Sloot, P., et al. eds. *Proceedings of Computational Science, ICCS 2004, Krakow, Poland, June 6−9, 2004. Part II*. Springer, Berlin, 65−72. Lecture Notes in Computer Science no. 3037. [http://algorithmicbotany.org/papers/solving.iccs2004.pdf]

Karwowski, R. and Prusinkiewicz, P., 2003. Design and implementation of the L+C modeling language. *Electronic Notes in Theoretical Computer Science,* 86 (2), 1-19. [http://algorithmicbotany.org/papers/l+c.tcs2003.pdf]

Minchin, P.E.H., Thorpe, M.R. and Farrar, J.F., 1993. A simple mechanistic model of phloem transport which explains sink priority. *Journal of Experimental Botany,* 44 (5), 947-955.

Nobel, P.S., 2005. *Physicochemical and environmental plant physiology*. Elsevier, Amsterdam.

Parter, S., 1961. The use of linear graphs in Gauss elimination. *SIAM Review,* 3 (2), 119-130.

Press, W.H., Teukolsky, S.A., Vetterling, W.T., et al., 1992. *Numerical recipes in C: the art of scientific computing*. 2nd edn. Cambridge University Press, Cambridge. [http://www.nrbook.com/a/bookcpdf.php]

Ramakalyan, A., 2005. *Linear circuits: analysis and synthesis*. Oxford University Press, Oxford.

Room, P.M., Maillette, L. and Hanan, J.S., 1994. Module and metamer dynamics and virtual plants. *Advances in Ecological Research,* 25, 105-157.

Salisbury, F.B. and Ross, C.W., 1992. *Plant physiology*. Wadsworth, Belmont.

Smith, K.C., Magnuson, C.E., Goeschl, J.D., et al., 1980. A time-dependent mathematical expression of the Münch hypothesis of phloem transport. *Journal of Theoretical Biology,* 86, 493-505.

Thompson, M.V. and Holbrook, N.M., 2003. Application of a single-solute non-steady-state phloem model to the study of long-distance assimilate transport. *Journal of Theoretical Biology,* 220, 419–455.

Thornley, J.H.M. and Johnson, I.R., 1990. *Plant and crop modelling: a mathematical approach to plant and crop physiology*. Clarendon Press, Oxford.

APPENDIX: AN ALTERNATIVE APPROACH AND ITS LIMITATIONS

A special case of the ladder circuit from Figure 1c can be solved without using the folding and unfolding operations. To see this possibility and realize its limitations let us assume that:

- all internal resistances r_{pk} of sources and sinks are equal to 0; and
- the electromotive forces associated with sources or sinks are proportional to the accumulated charges: $v_k = q_k/C_k$, where C_k is the capacitance of the source or sink associated with metamer k.

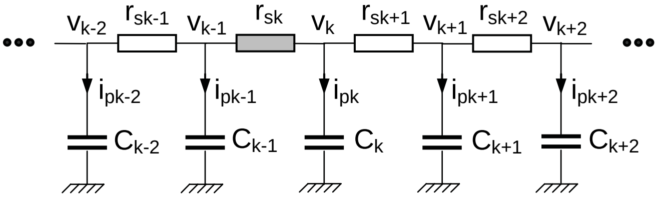

***Figure A1**. Special case of the circuit from Figure 1c*

The resulting circuit is shown in Figure A1. The current i_{pk} flowing into the capacitor C_k can be calculated as

$$i_{pk} = \frac{v_{k-1} - v_k}{r_{sk}} - \frac{v_k - v_{k+1}}{r_{sk+1}} \qquad \text{(A1)}$$

This current is related to the change of charge q_k accumulated in the capacitor C_k:

$$i_{pk} = \frac{dq_k}{dt} = C_k \frac{dv_k}{dt}. \qquad \text{(A2)}$$

By combining equations A1 and A2, we obtain:

$$\frac{dv_k}{dt} = \frac{1}{C_k}\left(\frac{v_{k-1} - v_k}{r_{sk}} - \frac{v_k - v_{k+1}}{r_{sk+1}}\right). \qquad \text{(A3)}$$

Given the initial distribution of the node potentials, we can thus calculate the progression of potentials in the sinks over time (and hence the accumulated charges) using the Euler integration method (Press et al. 1992):

$$v_k^{t+\Delta t} = v_k^t + \frac{1}{C_k}\left(\frac{v_{k-1}^t - v_k^t}{r_{sk}} - \frac{v_k^t - v_{k+1}^t}{r_{sk+1}}\right)\Delta t\,. \qquad \text{(A4)}$$

While tempting in its simplicity, this approach fundamentally relies on the assumption that the internal resistance r_{pk} of source or sink k (Figure 1c) is equal to zero. Without this assumption, the node potential v_k at any time t can no longer be presumed equal to the electromotive force e_k, and depends instead on all three potentials $v_{k\text{-}1}$, v_{k+1} and e_k. Since the potentials $v_{k\text{-}1}$, v_{k+1} are themselves not known, and this reasoning applies to all elements of the ladder, a system of n equations with n unknowns $v_1,v_2,...,v_n$ has to be solved. The folding and unfolding operations presented in Section "*Solving ladder and branching ladder circuits*" offer a convenient method for solving this system.