Published in final edited form as:

Comput Methods Biomech Biomed Engin. 2014 ; 17(14): 1519–1534. doi:
10.1080/10255842.2012.753066.

Incorporating measured valve properties into a numerical model of a lymphatic vessel

C.D. Bertram^a, C. Macaskill^b, and J.E. Moore jr.^c

C.D. Bertram: c.bertram@sydney.edu.au; C. Macaskill: c.macaskill@maths.usyd.edu.au; J.E. Moore: jmoorej@bme.tamu.edu

^aSchool of Mathematics and Statistics, University of Sydney, New South Wales 2006, Australia, ph: +61-2-9351-3646, fx: +61-2-9351-4534

^bSchool of Mathematics and Statistics, University of Sydney, New South Wales 2006, Australia

^cDepartment of Bioengineering, Texas A&M University, College Station, TX 77843-3120, USA

Abstract

An existing lumped-parameter model of multiple lymphangions (lymphatic vascular segments) in series is adapted for the incorporation of recent physiological measurements of lymphatic vascular properties. The new data show very marked nonlinearity of the passive pressure-diameter relation during distension, relative to comparable blood vessels, and complex valve behaviour. Since lymph is transported as a result of either the active contraction or the passive squeezing of vascular segments situated between two one-way valves, the performance of these valves is of primary importance. The valves display hysteresis (the opening and closing pressure-drop thresholds differ), a bias to staying open (both state changes occur when the trans-valve pressure drop is adverse), and pressure-drop threshold dependence on transmural pressure. These properties, in combination with the strong nonlinearity that valve operation represents, have in turn caused intriguing numerical problems in the model, and we describe numerical stratagems by which we have overcome the problems. The principal problem is also generalised into a relatively simple mathematical example, for which solution detail is provided using two different solvers.

Keywords

lymph transport; lumped-parameter model; fold bifurcation; differential algebraic equation; fluid-structure interaction

1. Introduction

We have previously published (Bertram et al. 2011a) a description of a lumped-parameter model of a small lymphatic vessel, consisting of several lymphangions (contractile segments between valves). For each segment, the model coupled a first-order ordinary differential equation (ODE) of mass conservation relating inflow and outflow difference to diameter change with a Poiseuille equation of momentum conservation and a nonlinear constitutive relation between lymphangion diameter and transmural pressure. Use of the Poiseuille

formulation is justified by the small Reynolds number of flow in the microscopic vessels under consideration. Pumping was induced by imposing a sinusoidal waveform of active wall tension causing an extra term in the constitutive relation, with a phase difference between successive lymphangions. Valves between lymphangions were modelled with a logistic function causing high resistance when the pressure drop across the valve was adverse, and *vice versa*; see Figure 1. The resulting set of equations was solved in Matlab (The MathWorks Inc., Natick MA, USA) as an algebraic minimum-finding problem of dimension $2n$ nested inside a simultaneous-ODE problem of dimension n , where n was the number of lymphangions in the model. The Matlab routine LSQNONLIN was used for the algebraic problem; ODE45 was used for the differential equation system. The whole model was vectorised such that the same code could be used for varying numbers of segments; up to eight were tested. The model was shown to be robust, reaching a stable converged pumping cycle over a wide range of parameter value combinations.

The model has since been developed further in almost every respect. Many of these changes have been driven by new experimental data on the mechanical and physiological properties of lymphatic vessels. The set of parameters and parameter variations used in the published model was aimed at ensuring reasonable output values of diameters, pressures and flow-rates, but the individual parameter values were not explicitly based on measured values. Many exciting new data have recently appeared, in particular on the properties of the valves in small lymphatic vessels (Davis et al. 2011). Their measured behaviour turns out to be far from simple. The valves display hysteresis, switching from open to closed, and from closed to open, at different values of the trans-valve pressure drop. These threshold values are themselves subject to variation depending on the transmural pressure environment (or on the local vessel diameter—a current controversy concerns which is more fundamental). Closely linked to the valve properties is the constitutive relation for the passive vessel wall properties; this has now been shown (Rahbar et al. 2012) to be far more nonlinear than was anticipated on the basis of the known behaviour of equivalent-sized venules and arterioles. The purpose of this paper is to describe the unexpectedly interesting mathematical structure of the model when these behaviours were incorporated. This in turn forced the development of specialised solution methods, which have yet to be fully generalised for use with standard solver packages.

2. Methods

2.1 Non-valve-related

A similar modelling strategy to that used by Bertram et al. (2011a) was followed here. However several major changes have since been made to aspects of the model other than the valves, and it will be convenient to describe these first. These changes concern the constitutive relation, the number of pressure variables per lymphangion, and the method of solution.

For the model's constitutive relation, transmural pressure was defined as the pressure in the vessel, less that outside. The empirical passive constitutive relation (see Figure 2) consists of measurements of diameter vs. transmural pressure for the vessel bathed in calcium-free physiological saline. The vessel becomes much stiffer at a certain diameter, such that

distension of the vessel significantly beyond that diameter requires unphysiologically high distending pressures. Because the strain is characterised by increase of diameter, the data necessarily pertain only to positive values of transmural pressure where the vessel remains circular in cross-section. *In vivo*, much lymph pumping occurs as a result of the vessels being squeezed by adjacent tissues, such as skeletal muscles bulging during voluntary contraction. Such influences squash the vessels into a non-circular cross-section, as does negative transmural pressure (internal pressure less than external). Transiently negative transmural pressure occurs physiologically, and certainly can occur in the model during a pumping cycle, depending on parameter settings. Therefore the measured data must be extrapolated to cover negative transmural pressures, since a region of unspecified behaviour leaves a numerical model liable to unrealistic solutions, here potentially involving impossible negative diameters. A flexible vessel acted upon by sufficient negative transmural pressure collapses to non-circular cross-sections, with eventual contact of opposite sides of the vessel wall. Description of such shapes is beyond the present model, but can be summarised in a gradual stiffening (reduction in slope of the curve of cross-sectional area as a function of transmural pressure) as that pressure becomes more negative (Flaherty et al. 1972). The curve is chosen such that zero cross-sectional area would be achieved only at infinite negative transmural pressure. For the model, this information is translated into an equivalent curve of transmural pressure against 'diameter', even though diameter is fictitious once the real vessel departs from circular.

Thus creating the constitutive relation involves fitting the measured pressure-diameter data while grafting on a collapse function. The collapse function, based on self-similarity theory for a vessel with a dual-tear-drop cross-section (Flaherty et al. 1972), is an inverse cubic term. Fitting the measured pressure-diameter data demanded a sum of quadratic polynomial and exponential terms. Trial values of the various constants were guessed, and varied until an approximate fit was achieved, then final values were calculated by nonlinear least-squares residual minimisation using the Matlab routine LSQNONLIN. Further adjustments were made to allow the curve to be normalised in ways that allowed comparison with the constitutive relations used in the published model (Bertram et al. 2011a) and improved versions adopted subsequently.

After the publication of the original model, it was found (Bertram et al. 2011b) that more pressure variables per lymphangion were required to allow the realistic simulation of backflow through a prolapsing valve, something that was not tested originally. In the published model, internal pressure was only defined at each end of the lymphangion (p_1 at the inlet, and p_2 at the outlet). The instantaneous average of p_1 and p_2 , minus the external pressure p_e , defined the transmural pressure. A single Poiseuille relation related $p_1 - p_2$ to the instantaneous average of the inlet and outlet flow-rates, $(Q_1 + Q_2)/2$. The use of this average flow-rate avoided linking the pressure drop to a flow-rate variable which was spatially offset, as would have been the case if either Q_1 or Q_2 alone had been used. However, it was found that this scheme produced unrealistic outcomes when modelling prolapse (overt valve failure in the face of a large adverse pressure drop). A minimal necessary modification (Bertram et al. 2011b) was to define a third pressure variable p_m at the centre of the lymphangion, giving the total situation shown in Figure 3. This new

pressure is used as the internal pressure in defining the transmural pressure for the constitutive relation. Two separate Poiseuille relations link $p_1 - p_m$ to Q_1 , and $p_m - p_2$ to Q_2 , respectively. The number of time-dependent variables for each lymphangion (and the dimension of the minimum-finding problem given to LSQNONLIN) thus increased (but see below).

Along with this change, the model was reconstituted as a formal differential-algebraic equation (DAE) system, for solution by the appropriate Matlab routine ODE23T. Our former solution scheme in effect did the same thing but less efficiently. Recourse to LSQNONLIN was no longer needed, and the resulting numerical model was both faster and more robust. A DAE system is a combination of algebraic and differential equations requiring simultaneous solution. The process involves definition of an $m \times m$ 'mass matrix' which identifies the purely algebraic constraints to the solver, where m is the total number of equations. Viewed as a DAE system, the introduction of p_m increased the number of algebraic constraints, and thus the order of the problem (although see below), but no new differential equations were invoked.

2.2 Valve-related

The measured valve properties were used in the form of two curves of valve threshold pressure difference as a function of transmural pressure, one curve defining the opening threshold and the other the closing threshold. These curves were generated by trial-fitting then least-squares fitting the measured data points with an appropriately chosen function, using LSQNONLIN as described above. A suitable function combines a straight line and a fractional power-law. The fits permitted emulation of the experimentally measured valve properties (Davis et al. 2011). However, as with the constitutive relation, the measured data only covered positive transmural pressures. The initial curves generated, while satisfactory fits to the data, became singular at zero transmural pressure. Therefore modified curves, extrapolating the data to negative transmural pressures in a conservative fashion, were designed and fitted. These curves combine a straight line and an exponential function of the square of the independent variable (the transmural pressure); see Appendix A.

The adoption of the hysteretic and transmural pressure-dependent valve properties brings two inherent complications. The first is switching immediately after valve opening or closure to the use of the other curve to determine subsequent closure or opening respectively. For each valve, a binary variable was introduced to track the current state of the valve. The second complexity is that the definition of transmural pressure also has to be changed immediately after switching, since in the experiments the controlling transmural pressure was that on the upstream side for opening, and on the downstream side for closing. Particularly when a valve is closed, the pressure difference across the valve can be large, and therefore the transmural pressure defined in terms of upstream pressure can be entirely different from that defined with downstream pressure. Again the binary state variable controls which definition of transmural pressure is operative.

With these changes, emulation of all the experimental valve tests was relatively easy, because the circumstances of these tests involved upstream and downstream pressures being varied slowly and explicitly under the operator's control. However, the inclusion of such

valves in the model for multiple lymphangions demands that the numerical model be able to survive unpredictable discontinuities in time in the solution state when each valve switched, with the pressures on either side of the valve now being outcomes of the model rather than controlled inputs.

To simplify the problem initially, we considered a model of a single pumping lymphangion, with constant-pressure reservoirs beyond the inlet and outlet valves (at pressures p_a and p_b respectively, where $p_b > p_a$), as in Fig. 3. This brings about the very considerable simplification that the pressure drop across each valve now depends on one pressure time-variable only. We found that the problem can be formulated as integration of a single first-order nonlinear ODE in the time-varying lymphangion diameter D ,

$$\frac{1}{D^3} \frac{dD}{dt} = \frac{p_1(D, t) - 2p_m(D, t) + p_2(D, t)}{32\mu L^2}, \quad (1)$$

subject to only two algebraic constraints,

$$p_1 = p_m(D, t) + \frac{64}{\pi} \frac{\mu L}{D^4} \frac{p_a - p_1}{R_V(p_1; p_a, p_e)}, \quad (2)$$

$$p_2 = p_m(D, t) - \frac{64}{\pi} \frac{\mu L}{D^4} \frac{p_2 - p_b}{R_V(p_2; p_b, p_e)}, \quad (3)$$

$$\text{where } p_m = p_e + f_p(D) + f_a(D, t). \quad (4)$$

Because of the implicit form of the constraints (2) and (3), direct substitution into (1) provides no simplification. Equations 2 and 3 both involve the highly nonlinear valve resistance function R_V shown in Fig. 1. Equation 4 involves the nonlinear passive constitutive relation $f_p(D)$, and the time-varying value of the term $f_a(D, t)$ originating in the active circumferential wall tension due to muscle contraction. These functions are defined algebraically in Appendix A. Equation 4 is not regarded as a separate algebraic constraint, but simply as convenient notation for a quantity entirely determined by the variables D and t . Thus the model order effectively reduces back to what it was prior to the introduction of p_m .

Early in the course of investigating how to deal with the discontinuities inherent in the solution of these equations, we also built single-lymphangion models which isolated the two possible valve characteristics. In one, the opening characteristic was used to determine the threshold pressure for a valve state change, irrespective of whether that state change was from closed to open or *vice versa*. In the other such model, the closing characteristic was used for both valve opening and valve closure.

However, the model adaptations for realistic valve behaviour caused the previously successful DAE scheme to fail. Matlab's DAE solution scheme, based on use of either

ODE23T or ODE15S, is built around the concept of maintaining a given accuracy by reducing the time-step as necessary. This caused the scheme to exit at the first attempted valve switching, with the time-step having been reduced to the minimum possible, irrespective of which solver was used. We investigated other DAE solver packages, in particular the Sundials package (SUIte of Nonlinear Differential ALgebraic equation Solvers) from Lawrence Livermore National Laboratory, Livermore CA, USA (see <https://computation.llnl.gov/casc/sundials/>), but found the same problem; the concept of adapting the time-step to maintain the accuracy was fully embedded.

To get over this hurdle we use a fourth-order-accurate Runge-Kutta integrator where we can control the time-step, keeping it constant across the discontinuity of a valve switching to the other state. We also identified a second problem, in the form of the algebraic constraint equations being solved for p_1 and p_2 . Since in the single-lymphangion model these are one-dimensional functions, they can readily be inspected graphically at each time-step, and indeed animated as time-varying curves for which the intersection with zero is required. It was found that, depending on the values of the parameters controlling the transmural pressure at which the valve was working, these functions could have multiple roots. As shown schematically in Figure 4b, the function, which incorporated the smooth but rather step-like valve resistance change across the threshold pressure drop, could include a region of gradient of opposite sign from that of the rest of the curve. When this region intersects the horizontal axis there are three equation roots; when it does not there is one. The functions themselves are shown in the Results section.

A closely related problem occurs when the above region ceases to intersect the axis, as in Fig. 4c. We initially opted to use Matlab's FSOLVE (with Levenberg-Marquardt algorithm (Moré 1977) option) to perform the iterative root-finding. This fails when it converges to a point that is not a root but instead a local maximum or minimum, such as is offered every time when the triple-valued function 'lifts off'. With this knowledge in hand, it is possible to specify an appropriate alternative search start-point when no root is found, based on the direction of function movement at the time, i.e. whether the function was heading upwards or downwards. However, this tailoring of FSOLVE's search behaviour for a given operating point potentially leads to unreliability for other operating points.

In view of these difficulties with FSOLVE, we turned to the alternative Matlab root finder FZERO, which uses Dekker's method of combining bisection, secant, and inverse quadratic interpolation methods (Forsythe et al. 1976). This is much faster, and never gets marooned on local non-root extrema, but sometimes causes the model to return less accurate results, despite FZERO's tiny default tolerance.

Having achieved success with our fixed-time-step Runge-Kutta ODE solver allied to either FSOLVE or FZERO, we also tried using LSQNONLIN to find the global minimum of the two-dimensional surface offered at each time-step by the orthogonal combination of the functions for p_1 and p_2 . Rather to our surprise, LSQNONLIN was successful, giving overall solution times intermediate between those of FZERO and FSOLVE. However, there was the same intermittent buildup of substantial error (as flagged by both minimiser residuals and our own consistency checks) as was exhibited by FSOLVE.

Again close examination showed that the root was not being identified when three roots had just given way to one, requiring the solution to jump to a new branch. In order to avoid the computational penalty of doing a global search at every time-step, the algorithm employed continues to search locally for a while. Only when this fails and the accumulated error has become large is a global search initiated which finds the new branch. This causes the solution to jump later than it should have, and to incur large errors until it does so. A procedure was devised to overcome this fault, consisting of the following steps. Noting the first time-step at which large residuals were generated, the solution process is allowed to proceed for as many time-steps as are necessary for the algorithm to find and jump to a trajectory approximating the new branch. Using this as the new trial solution, a return to the earlier time is then forced. The correct solution is thereby identified at the time-step immediately after the jump. The solution is then allowed to proceed normally, in the process replacing the previously found erroneous post-jump trajectory with the new correct one. The procedure is described in more detail below.

Finally we show that it is possible to combine LSQNONLIN with Matlab's ODE45 general-purpose ODE solver. Although ODE45 organizes its own time-steps internally (while presenting a solution at the specified intervals), this combination does not enforce overall the approach of ensuring a fixed accuracy by time-step reduction. The pairing can therefore be likewise constrained to return to the site of a jump and overwrite the errors incurred before finding a new solution branch. The resulting solution process is potentially faster than our fixed-time-step Runge-Kutta solver. We also show that either the LSQNONLIN/ODE45 combination, or LSQNONLIN with our fixed-time-step Runge-Kutta solver, can be used to solve the equations for two lymphangions in series between fixed-pressure reservoirs.

3. Results

Fractional power-law relations provided a good fit to the experimental data (Davis et al. 2011) on the dependence of valve-opening and valve-closing pressure-drop thresholds on the transmural pressure; see Figure 5. Details of the curve coefficients are given in Appendix A. With these curves controlling the valve resistance function of Fig. 1, we could emulate the results of all the experimental tests applied by Davis et al. (2011) to excised lymphatic vessel segments containing a valve; an example of such a simulation is shown in Figure 6. The procedure here involves varying the valve outlet pressure with time as shown in the top trace, and corresponds to that used for their fig. 3 by Davis et al. (2011). Approaching each change of state, the valve resistance varies with the proximity of the current valve pressure drop to the current valve switching threshold, as determined by (1) which state it is currently in, and (2) what the current transmural pressure is. The valve resistance gradually moves towards the mid-transition point of Fig. 1. When the conditions for valve switching are attained, two discontinuous changes occur: the characteristic used to determine the threshold is swapped for the other one, and the definition of transmural pressure changes, in accordance with the abscissae of Fig. 5. As a result, the valve now finds itself operating far from the transition point, and the valve resistance changes discontinuously to complete the switching. These events are shown in more detail in the manoeuvre of Figure 7.

Figure 8 shows the form of the functions of p_1 and p_2 which require iterative solution at each time-step for p_1 and p_2 when the valves are open and therefore about to close. Normally, when the inlet valve is open, the outlet valve is closed, and *vice versa*, but in this case the closing characteristic was being used uniquely, to determine the switching threshold for both opening and closing. The function, which is $f(x)$ when eq. 2 or 3 is rewritten as $f(x) = 0$, consists of three terms, or summed components. Taking that for p_1 (eq. 2) as an example, and referring to the left-hand panel of Fig. 8, the first component is a fixed sloping line which represents the possible values of p_1 itself. The second component is a horizontal line representing the current value of $-p_m$; the level of this component varies slowly with D and t according to eq. 4. The third component is the function of Q_1 and D which also contains the valve resistance function R_V with its recognisable smoothed-step nonlinearity. This component has a factor $p_a - p_1$, and therefore changes sign at the trial value $p_1 = p_a$. Its magnitude is large when D is small, and *vice versa*. Thus the whole function (solid line) slowly moves up and down in time, and the size of the embedded step varies in time also. Together these behaviours caused the function to intersect the horizontal axis (presenting a solution for p_1) at either one or three locations.

Equally good fits to the measured valve-opening and valve-closing data were achieved when the function used to fit the data was of a form that could be continued into the region of negative transmural pressure. Figure 9 shows these fits; again the coefficients are specified in Appendix A. This extrapolation allowed the successful operation of a single-lymphangion model based on valves using the opening characteristic for both opening and closing. Figure 10 demonstrates this, showing waveforms of pressure, diameter and flow-rate for this model during one cycle of fully periodic pumping. The time-varying resistance of the two valves is also shown; note that it varies smoothly and continuously, including when the mid-resistance has been passed. This behaviour contrasts with what follows.

The reason for the success of this model was that the functions (algebraic constraints) to be solved for the pressures p_1 and p_2 always remained single-valued, as is shown in Figure 11. For the equivalent model using the valve-closing characteristic, as shown in Fig. 8, it was necessary to provide model-specific code to avoid FSOLVE getting stuck at a local extremum. With this in place, a value for each of the pressures could always be found, and when the situation of Fig. 4b pertained, there were three possibilities. Figure 12 shows the resulting waveforms of p_1 and p_2 , including all three values where these differed. The curves exhibit classical fold bifurcations, and this is the underlying dynamic of the three roots, which were present briefly during the pressure rise (when the inlet valve closed and the outlet valve closed) and for a longer period as intra-lymphangion pressure decreased during the decay of active tension and each valve reverted to its original state.

When three roots exist, which is the correct one? The software continued to follow the root it had been tracking before others appeared, and this is the correct solution; only when the existing root ceases to exist at a fold bifurcation is it both necessary and appropriate to jump to the new solution, which by that time is unique.

Once the closing characteristic had been mastered, the assembly of a single-lymphangion pump using both the opening and closing characteristics to determine these respective valve

state changes presented no further problems. At a jump between valve characteristics, the waveforms of pressure and of flow-rate exhibited discontinuities, whereas only the slope of the diameter waveform was discontinuous. This was true for valve opening as well as valve closure, the reason on opening being not a discontinuity in the sense shown in Fig. 12 but that caused by switching to the other valve characteristic to determine the next change of state. In terms of valve resistance the result was to cause valve resistance to vary smoothly until it passed the mid-value, then to jump to a new value which was invariably at or very close to one of the extremes R_{Vmin} or $R_{Vmin} + R_{Vmax}$.

Using FZERO instead of FSOLVE, the measures introduced to guide FSOLVE past local extrema were no longer necessary. Solution with FZERO was over ten times faster (albeit still 2.5 times slower than the formerly used DAE solution under circumstances where this could be compared)¹.

Both FZERO and FSOLVE are specific to the task of finding roots of one-dimensional functions. Neither was therefore suitable for the more general problem posed by a pump consisting of more than one lymphangion, where the algebraic constraints on the several pressure variables have to be solved simultaneously. The next step was to attempt simultaneous solution of at least the two-dimensional problem posed by a single lymphangion, using LSQNONLIN. This was more successful, but still gave rise to short periods of large solution residuals. Comparison with the previous solutions showed that these occurred when three roots gave way to one. The solution using LSQNONLIN took almost six times longer than with FZERO, but was still faster than using FSOLVE¹.

The combination of LSQNONLIN for the algebraic constraints and the fixed-time-step Runge-Kutta integrator for the differential equations was then applied to the problem of two lymphangions in series. In this case the valve between the two lymphangions experiences time-varying pressures on both sides, and simultaneous solution of the four-dimensional problem involving all the algebraic constraint equations is essential. Again the method proved equal to the task, but again there were episodes of large residuals from LSQNONLIN during each pumping cycle, and these appeared linked to the jumps to a new root after a fold bifurcation.

The occurrence of large residuals was investigated, and their origin found. This condition indicates that there are no longer three solutions satisfying the constraint, but rather one solution and one local minimum, close to zero. The difficulty for the nonlinear solver is that we have a starting solution (at the new time point) close to the local minimum, which does not correspond to a root. This problem is exacerbated as time increases. Eventually, the local minimum is so far from zero that the solver (LSQNONLIN) jumps to the correct branch. However, all the intermediate time results with large residuals are clearly incorrect. The solution is to store the current (correct solution), go back to the time where residuals first became large (the bifurcation point) or just past this, and use the stored solution as a starting point. As this is on a trajectory close to the appropriate branch we get convergence to the

¹These tests were conducted using Matlab release R2010a, in which FSOLVE was handicapped by unnecessary internal function calls. This problem has been rectified by The Mathworks in Matlab release R2011b.

required solution easily (and a jump at the bifurcation point, replacing the previously found delayed jump). Integration can then proceed with small residuals until the next such occurrence. Figure 13 shows a cycle of two-lymphangion pumping with the back-tracking implemented. The instants when valves change state are marked by discontinuities of flow-rate and pressure, and also (but less obviously) by discontinuities of diameter waveform slope.

Our ultimate aim is to find a way to use an existing DAE system solver on problems with these properties. As a step toward that we next investigated the use with LSQNONLIN of Matlab's general-purpose ODE solver ODE45, using code that incorporated the back-tracking to correct instances of late jumping and large residuals. Provided that the interval over which ODE45 was deployed did not contain more than one jump, it proved possible to obtain accurate solution of the two-lymphangion model by this means. We then substituted the solver ODE23T for solution of the differential equations, while retaining LSQNONLIN to solve the algebraic constraint problem. The solver ODE23T is designed to solve entire DAE systems, but was thus here used only to solve the ODE. Again subject to the same caveat, i.e. that the integration interval contain no more than one jump, solution was again possible. In the absence of advance information about where the jumps were going to occur, the ODE solver was deployed in a loop which defined the integration interval as equal to the time-step for results output (allowing the solver to organize its own time-stepping only within this small interval). This is inefficient, and we intend to seek a more efficient scheme.

The solutions with ODE23T and with our own fixed-time-step Runge-Kutta solver were almost indistinguishable, although our internal consistency checks revealed more frequent small errors in the ODE23T solution. In both cases, the great majority of solution time was spent in LSQNONLIN, emphasizing the need to achieve a true DAE solution without use of a separate algebraic-constraint solver.

4. Discussion

Although this paper has been concerned with a model of a specific biological system, the mathematical problem posed can be generalised. A realisation of the generalised problem is presented in Appendix B.

As alluded to above, our procedure for dealing with the jumps is inefficient. The procedure requires that there be a maximum of one jump in the interval of time over which the ODE solver is let loose. An entire cycle of steady-state pumping for a single lymphangion, extending over a few seconds of simulated time, is likely to contain only four jumps, corresponding to the opening and closing of each of the two valves. But the closure of the inlet valve and the opening of the outlet valve could occur in successive result-output time-steps. Consequently the length of this time-step is currently the limit on the length of the time interval allowed to the ODE solver. There are hundreds of time-steps per pumping cycle, so this severely restricts the ODE solver's ability to optimise its approach. The problem is not fatal, but it is a known reason why the procedure is computationally expensive.

We have thus solved many of the problems posed by the incorporation of the physiological data into the series-lymphangions model, but there is still some distance to go before we achieve our goal of a vectorised DAE solution. Only at that stage will the model be ready to use as an element in a wider network model of part of the whole lymphatic system, this being our ultimate aim.

We have here highlighted the intriguing mathematical problems encountered in the further development of an existing lumped-parameter model to encompass realistic parameter values and functional forms, in particular relating to the adoption of hysteretic and transmural pressure-dependent valves. Relative to the originally published model (Bertram et al. 2011a), other changes have also been made: a different pressure-diameter relation, one more pressure variable per lymphangion, and a refractory period between contractions. In a related paper (Bertram et al. 2012) we report further on the findings for the application of interest, namely simulation of pumping by a lymphatic vessel.

Acknowledgments

CDB thanks Dr Peter Schmid of the Laboratoire d'Hydrodynamique (LadHyX), Ecole Polytechnique, France, for many useful discussions, and is grateful for the facilities afforded to him during a sabbatical year at LadHyX. JEM gratefully acknowledges support from U.S. National Institutes of Health grant R01-HL-094269.

References

- Bertram CD, Macaskill C, Moore JE jr. Simulation of a chain of collapsible contracting lymphangions with progressive valve closure. *ASME J Biomech Eng.* 2011a; 133(1):011008-011001–011008-011010.10.1115/1.4002799
- Davis MJ, Rahbar E, Gashev AA, Zawieja DC, Moore JE jr. Determinants of valve gating in collecting lymphatic vessels from rat mesentery. *Am J Physiol (Heart Circ Physiol)*. 2011; 301(1):H48–H60.10.1152/ajpheart.00133.2011 [PubMed: 21460194]
- Rahbar E, Weimer J, Gibbs H, Yeh AT, Bertram CD, Davis MJ, Hill MA, Zawieja DC, Moore JE jr. Passive pressure-diameter relationship and structural composition of rat mesenteric lymphangions. *Lymphatic Research and Biology*. 2012 in press.
- Flaherty JE, Keller JB, Rubinov SI. Post-buckling behaviour of elastic tubes and rings with opposite sides in contact. *SIAM Journal of Applied Mathematics*. 1972; 23:446–455.
- Bertram, CD.; Macaskill, C.; Moore, JE, jr. Effects of lymphangion subdivision in a numerical model of a lymphatic vessel. Paper presented at: Proceedings of the ASME 2011 Summer Bioengineering Conference (SBC2011); Farmington, Pennsylvania, USA. 2011b.
- Moré, JJ. Numerical Analysis (Lecture Notes in Mathematics 630). Springer Verlag; 1977. The Levenberg-Marquardt algorithm: implementation and theory; p. 105-116.
- Forsythe, GE.; Malcolm, MA.; Moler, CB. Computer Methods for Mathematical Computations. Prentice-Hall; 1976.
- Bertram CD, Macaskill C, Davis MJ, Moore JE jr. Development of a model of a multi-lymphangion lymphatic vessel incorporating realistic and measured parameter values. *Biomechanics and Modeling in Mechanobiology*. 2012 re-submitted.

Appendix A

The measured valve opening and closing data were initially fitted with equations of the form

$$\Delta p_o = a_1 + a_2 \Delta p_{tm} - a_3 (\Delta p_{tm} / a_4)^{a_5}, \quad (\text{A.1})$$

where p_o is the pressure difference $p_{in} - p_{out}$ across the valve at the midpoint of its resistance transition, and for opening $p_{tm} = p_{in} - p_e$, $a_1 = 2.962$, $a_2 = 0.00346$, $a_3 = 3.235$, $a_4 = 9.960$ and $a_5 = 0.0388$; for closing $p_{tm} = p_{out} - p_e$, $a_1 = 1.541$, $a_2 = 0.03201$, $a_3 = 3.772$, $a_4 = 9.860$ and $a_5 = 0.2135$ (with p_o and p_{tm} in units of cmH₂O); p_{in} and p_{out} are the pressure at valve inlet and outlet (defined in the direction that the valve is set up to permit flow). These are the fits shown in Fig. 5. To enable extension to negative values of p_{tm} , the data were later fitted with equations of the form

$$\Delta p_o = b_1 - b_2 \Delta p_{tm} - b_3 \left(1 - \exp \left(-b_4 \Delta p_{tm}^2 \right) \right), \quad (A.2)$$

where for opening $p_{tm} = p_{in} - p_{ext}$, $b_1 = 0.1407$, $b_2 = 0.01060$, $b_3 = 0.2465$ and $b_4 = 0.6728$, and for closing $p_{tm} = p_{out} - p_{ext}$, $b_1 = -0.1274$, $b_2 = 0.04790$, $b_3 = 1.208$ and $b_4 = 0.4354$. These are the fits shown in Fig. 9.

The measured pressure-diameter data constituting the required passive constitutive relation were fitted by a function having the form

$$f_p(D) = P_d \left[c_1 \left(\frac{D}{c_9} - c_2 \right)^2 + c_3 \exp \left(c_4 \left(\frac{D}{c_9} - c_5 \right) \right) + c_6 + c_7 \left(\frac{D}{c_9} - c_8 \right) + c_{10} \left(\frac{c_9}{D} \right)^3 \right]. \quad (A.3)$$

The values of the constants are given in a related paper (Bertram et al. 2012). The last term was not needed to fit the data, but was added to provide sensible behaviour at negative transmural pressure where there were no data. The remaining constants were adjusted accordingly.

Active contraction was created in the model by the addition to the constitutive relation of a term

$$f_a(D, t) = \frac{M_0}{D} [1 - \cos(2\pi f(t - t_c))], \quad t_c \leq t < t_c + 1/f, \quad (A.4)$$

where M_0 is the peak active tension, and a contraction lasts from $t = t_c$ to $t = t_c + 1/f$. A further parameter t_r defined the time between contractions (the refractory period).

Appendix B

As a simple test case, look for the solution of the ODE

$$\frac{dy}{dt} = x \quad (B.1)$$

where $y(0)$ is given, and subject to the constraint

$$f(x, t) = 0, \quad (B.2)$$

where

$$f(x, t) = 36\sin t - 10 + (2\sin t - 5)x + x^3, \quad (\text{B.3})$$

which implicitly determines x for any given t .

The system (B.1–3) is a DAE, and there are many routines that can deal with such problems, directly incorporating the constraint. But when there is a jump in the value of x , these routines have difficulty. If the jump point can be identified explicitly, then one can integrate up to the jump using a standard DAE solver. Assuming that the dependent variable (y here) is continuous, one can then re-initialize (these solvers need to estimate consistent initial values of the derivative dy/dt) and continue. However, when there are many such jumps, occurring at locations that can only be determined as the equations are integrated, control of the DAE solver becomes tricky. In particular, the presence of a jump will usually be signalled by the routine taking smaller and smaller time increments and eventually reporting that it cannot progress beyond some time-point t_0 . Obviously one can then recalculate the solution from the initial time-point up to a time just before the failure time, taking this to be the appropriate jump point. But this whole approach is typically very inefficient. Alternatively we can solve the ODE (eq. B.1) with, for example, a simple Runge-Kutta method, so long as the value of the right-hand side of the ODE can be calculated at any time t , which in this case requires an appropriate value of x satisfying the constraint (B.2). A nonlinear solver such as the Matlab routine LSQNONLIN can be used for this, but will also have difficulty at the jump points, as described for the test example below.

Usually x should be continuous, except at bifurcation points where the zero of $f(x, t)$ that is being followed vanishes. The example right-hand-side function x and the associated constraint (B.2) have been chosen so that at some times t there are three solutions satisfying the constraint, while at others there is only one. In Figure B.1 a set of typical $f(x, t)$ curves is shown for a number of different t values. As the zero of $f(x, t)$ with maximum x is tracked in time, eventually it will coalesce with the adjacent zero, for t just less than 0.4, and then at this bifurcation point both zeros will vanish, forming instead a positive local minimum. It is at this stage that we must jump to the one remaining zero, thus giving a jump in x ; the difficulty is to find a good starting guess for the root-finder, as many such routines will rather find the local minimum near $x = 1$. One strategy that works with this test-case (but not all such test-cases) and for our lymphangion equations is the following.

When solving for x as a root of $f(x, t)$ using, e.g., LSQNONLIN, the residual (i.e. a suitable measure of the extent to which the constraint $f(x, t) = 0$ is not satisfied) can be monitored. Clearly if the local minimum is returned rather than a true root, then the residual will grow as t increases beyond the bifurcation point. This behaviour can be used as a trigger to initiate a search for an alternative (true) zero. The method proposed here is incrementally (and temporarily) to increase the value of t , within the right-hand-side function, which typically leads to an increase in the residual, corresponding to the movement of the local minimum away from the x -axis. Eventually, the nonlinear solver should move away from this minimum (as the residuals become ever larger) and move to a true zero (e.g. at $t = 0.5$ and x

–3 in Fig. B.1). The value of x so determined can then be used as an appropriate first guess for iterating to find x at the original time t at which the bifurcation was first signalled by the presence of large residuals. Under some circumstances, it may be necessary to provide a small random perturbation to this initial guess in order to determine the new root successfully.

The way in which this works for our test example is illustrated in Figure B.2. Consider for example Fig. B.2a. Here a fixed time-step ($\Delta t = 0.01$) fourth-order Runge-Kutta method is used. Consider, for example, the third jump, near $t = 7$. We see that x jumps from a value just greater than one to a value somewhat less than –2. The presence of a jump is indicated when the nonlinear solver, in an attempt to find x in the right-hand-side routine, encounters a large residual, greater than a pre-set threshold, at some time, say t_0 . The value of t is then incremented successively by a small quantity (here equal to the time-step), with the corresponding output x indicated by the black dots. For all these cases the residuals are unacceptably large. Eventually, a jump in x takes place, to the value indicated by the red circle, with the residual now below the threshold. We can in fact with this particular example now continue integrating in time from this new x value, as the right-hand-side function does not depend explicitly on y as it does in the lymphangion equations. However, even here, in order to resolve the jump adequately in time, it is necessary to return to the value t_0 , and use the current value of x as an initial guess for the nonlinear solver. Once the appropriate value of x has been found, indicated by the blue dot, the new value of the right-hand-side function is known, and the integration can continue.

A similar procedure can be used with the Matlab routine ODE23T replacing the fixed time-step Runge-Kutta code; see Fig. B.2b. Here ODE23T is used purely as an ODE solver; none of its DAE-solving ability is employed. It is also necessary to divide the overall time interval (here, 20 units) into suitable sub-intervals (here of length 0.2), with a separate call to ODE23T for each of the sub-intervals. This last requirement is necessary to avoid problems with multiple jumps, as the Matlab solver does not always advance forward directly in time, but sometimes backtracks; this behaviour causes problems with the present approach if multiple jumps are found in a single call to the solver. We see that the results with this code are very similar to those found with the simple Runge-Kutta solver, although sometimes the number of iterates (indicated by black dots in Figs. B.2a and B.2b) required before the new branch is found is greater with ODE23T. Since variable time-step integrators sometimes backtrack in time, spurious values of x may be obtained at times just prior to a jump (see the detail in Figure B.3). These are true zeros, but do not correspond to the branch being followed before the jump. There seems to be no way to overcome this problem (the root found will typically be that closest to the previous value of x which is used as the initial guess), but this behaviour does not seem to affect the ability of the variable-time-step integrator to solve the ODE, as indicated by the good comparison in Figure B.4.

Figure 1.

The function describing normal valve operation is specified by four parameters: maximum and minimum resistance, the pressure drop across the valve at the midpoint of the transition, and the steepness of the transition slope. Positive p here means a pressure difference which is favourable to forward flow through the open valve. For the more complex valve characteristics described in this paper, the mid-transition pressure parameter p_o becomes a function of valve state and transmural pressure.

Figure 2.

The left panel shows (in green) the measured pressure-diameter data, overlaid by the fitted curve (red) with added collapse function. The diameter data are normalised by the value 259.8 μm , which occurred at a transmural pressure of 5 cmH₂O. The right panel shows how the curve thus normalised can be rescaled to represent lymphatic vessels of any desired diameter and stiffness.

Figure 3.

The time-variables involved in the solution of a single-lymphangion pump are the pressures p_1 , p_m and p_2 , the flow-rates Q_1 and Q_2 , and the diameter D . The reservoir pressures p_a and p_b ($p_b > p_a$) are fixed, as is the external pressure p_e . The valve resistances depend on the local p ($= p_a - p_1$ for valve 1 here) as shown in Fig. 1.

Figure 4.

Schematic of the form of the function of which a zero constituted the required pressure when using the valve-closing characteristic. The three panels illustrate a situation where the function is moving upwards as t increases. In (a), there is only one intersection with the abscissa, i.e. one root (marked ●); this is the required pressure. In (b), the function having moved up, there are three possible roots. In (c), the function having moved up further, there is again only one root, but it is possible for a solver which has previously been tracking the largest- x root to get stuck at the local minimum which is not a root (marked ○).

Figure 5.

Curves fitted to the measured opening (above) and closing (below) data for a rat mesenteric lymphatic vessel of around 275 μm maximum diameter. The valve is biased to stay open; it switches when the pressure drop across it is already negative (adverse), particularly in respect of closure. These biases become more pronounced when the transmural pressure is high. Note the change of transmural pressure definition between the two panels. Fig. 6a in the legend refers to fig. 6A of Davis et al. (Davis et al. 2011).

Figure 6.

A simulation emulating one of the tests applied to the lymphatic valves *in vivo*. The top panel shows the pressures p_{in} and p_{out} applied to the lymphatic segment on either side of the valve. Pressure p_{in} (in blue) was constant; p_{out} (in red) was varied in a series of increasing triangular peaks. The pressure drop applied across the valve was thus $p = p_{in} - p_{out}$ (shown in green). The resulting changes in valve resistance are shown in the lower panel.

Figure 7.

Events during a single cycle of valve closure then reopening in response to gradual variation of p_{out} (as shown in red). The smaller panels on the right show the time-course of valve resistance change in detail.

Figure 8.

The functions (solid blue curves) to be solved for the pressures p_1 and p_2 at a selected time-step in the single-lymphangion problem. Both functions exhibit a region of slope of opposite sign. This functional form was exhibited by the valve-closing characteristic (here used exclusively for both valves); the equivalent functions based on the valve-opening characteristic were monotonic. In the case of p_2 , there is nevertheless only one possible root at this time-step. For p_1 , there are three roots at this time-step. Also shown are the components (red, magenta and blue dotted curves) which together made up the function, the pressure difference (green dashed lines) which determined where $f(Q,D)$ changed sign, and the value of the currently found roots for p_1 and p_2 (vertical black dotted lines). See text for further explanation.

Figure 9.

Curves fitted to measured data for valve opening (above) and valve closing (below), to allow the model to continue if $p_{tm} < 0$. Note the changed definition of transmural pressure on the abscissa between the two panels. Fig. 6a in the legend refers to fig. 6A of Davis et al. (Davis et al. 2011).

Figure 10.

From top to bottom, waveforms of pressure, diameter, flow-rate and valve resistance during one cycle of pumping caused by the imposition of active tension following the time-course shown in the second panel (lighter trace). See panel legends and titles for identification of individual traces. The pump here employed valves using only the opening characteristic shown in the upper panel of Fig. 9, to determine both opening and closing.

Figure 11.

Showing (solid blue curves) the form of the functions which determined p_1 and p_2 (where they intersect the horizontal axis, i.e. $f(x) = 0$) in the single-lymphangion model with valves using the opening characteristic only. Unlike the equivalent curves in Fig. 8, these always remain single-valued. See caption to Fig. 8 for explanation of the other curves.

Figure 12.

The waveform of $p_1(t)$ (upper panels) and $p_2(t)$ (lower panels) during one cycle of pumping of the single-lymphangion model, with valves using the closing characteristic for both opening and closing. At certain times the solution for the pressure becomes multi-valued; the three roots then form a pair of fold bifurcations. The right-hand panels show detail of the small folds which occurred just after $t = 13$ s.

Figure 13.

A cycle of pumping for two lymphangions in series between constant-pressure reservoirs. At this particular operating point (combination of parameter settings), the middle valve stays open all the time once the steady pumping state has been achieved, but closes during the cycles of start-up transient before repetitive cycles are achieved.

Figure B.1.

Example right-hand-side function $f(x,t)$ for $t = 0.2, 0.3, \dots, 0.6$, showing the jump in the number of roots at time $t = 0.4$.

Figure B.2.

Time evolution of $dy/dt = x$ for (a) the fixed-time-step fourth-order Runge-Kutta method and (b) the variable time-step Matlab routine ODE23T. For both plots, at each jump the black dots indicate iterates stepping forward in time, triggered by the presence of a large residual indicating a jump, the red circle is the solution given by the root-finder once it has jumped to the correct branch (exact here where there is no dependence of the ODE right-hand side on the dependent variable y) and the blue dot indicates the recalculated x value just beyond the jump point t_0 .

Figure B.3.

Comparison of the numerical solutions found by the fixed-time-step Runge-Kutta method (blue) and the variable time-step Matlab solver ODE23T (red). Red circles show the values returned by calls to the right-hand-side routine when ODE23T is used. Inappropriate values of x occur prior to the jump.

Figure B.4.

Comparison of the numerical solutions found by the fixed-time-step Runge-Kutta method (blue) and the variable time-step Matlab solver ODE23T (red). The jumps in x found in Fig. B.2 are reflected in the discontinuities in the slope of y .