Computers and Chemical Engineering 50 (2013) 152–183

Contents lists available at SciVerse ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

Finite element solution of coupled-partial differential and ordinary equations in multicomponent polymeric coatings

Raj Kumar Arya*

Department of Chemical Engineering, Jaypee University of Engineering & Technology Guna, A. B. Road, Raghogarh, Guna 473226, M.P., India

ARTICLE INFO

Article history:
Received 11 July 2012
Received in revised form 25 October 2012
Accepted 31 October 2012
Available online 20 November 2012

Keywords:
Coupled partial differential equations
MATLAB solution
Multicomponent polymeric coatings
Free volume theory
Galerkin's method of finite elements

ABSTRACT

Mass transport equations in multicomponent polymeric coatings are nonlinear coupled partial differential equations. These equations were solved using Galerkin's method of finite elements which converts them to ordinary differential equations. Residuals were made orthogonal by using quadratic basis functions. Non-uniform elements were used to capture steep concentration gradient near the top of the coating. Finite element formulation has been solved using ode15s of MATLAB. Results are in very good agreement with the earlier results using different solution techniques.

1. Introduction

Diffusion is usually the rate controlling step during transport in many polymer solvent systems. Fick's law of diffusion, states that the flux of a diffusing component at a location equals the product of diffusion coefficient and its concentration gradient there. For polymer solvent systems, the diffusion coefficient is a strong function of temperature and concentration. Free volume theory (Vrentas & Duda, 1977a, 1977b) describes this function accurately for several systems involving one polymer and one solvent.

Flux of the solvent at a location in multicomponent systems is determined by not only its concentration gradient but also those of others. The diffusion coefficient of the solvent, which combines with its own concentration gradient is called main-term coefficient and those that combine with gradient for other solvents are called cross-term coefficients. In fact, to describe diffusion in a N-component system, $(N-1)^2$ mutual diffusion coefficients are needed.

Recently, a unified theory (Price & Romdhane, 2003) is proposed of which the existing theories are special cases. They showed that the theories predict almost same average concentration of solvent in a drying ternary coating. But, they could predict different concentration profile of the solvent inside the coating. Such profiles are important in coatings that phase separate during drying.

Transport equations in multicomponent polymer–solvent–solvent systems are non-linear coupled partial differential equations. This paper deals with the solution scheme and the optimization of free-volume parameter required for diffusion models.

2. Governing equations

Fig. 1 shows the schematic of a drying ternary coating that has been cast on impermeable substrate. As the solvent reaches the surface from the bottom, it evaporates into air. As solvents depart, the coating shrinks with time. There is no mass transfer through the substrate; hence flux of both the solvents is zero at the substrate. The coating is heated from both the top and bottom sides.

2.1. Mass transport

The rate of change of concentration of solvent 1 equals gradient of flux, which is due to its own concentration gradient and that of solvent 2.

---

* Corresponding author. Tel.: +91 7544 267051x140; fax: +91 7544 267314.
E-mail address: raj.arya@juet.ac.in

0098-1354/$ – see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.compchemeng.2012.10.015

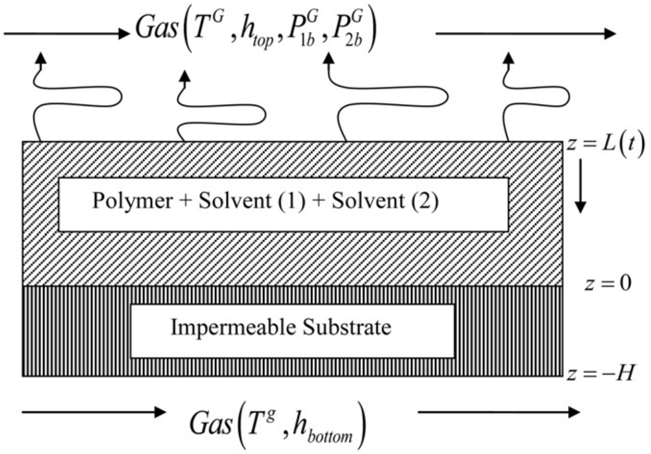


Fig. 1. Schematic of a drying coating.

Mass balance for solvent 1:

$$\frac{\partial c_1}{\partial t} = \frac{\partial}{\partial z}\left(D_{11}\frac{\partial c_1}{\partial z}\right) + \frac{\partial}{\partial z}\left(D_{12}\frac{\partial c_2}{\partial z}\right) \tag{1}$$

Mass balance for solvent 2:

$$\frac{\partial c_2}{\partial t} = \frac{\partial}{\partial z}\left(D_{21}\frac{\partial c_1}{\partial z}\right) + \frac{\partial}{\partial z}\left(D_{22}\frac{\partial c_2}{\partial z}\right) \tag{2}$$

The reference velocity is chosen to be volume average velocity because it is shown to be equal to zero if there is no change in volume on mixing [4].

c_i, is the concentration of solvent i, t is the time, z, is the thickness of the coatings at anytime, D_{11} and D_{22} are main-term diffusion coefficients that characterize transport due to solvents own concentration gradient, D_{12} and D_{21} are cross-term diffusion coefficients that characterize transport due to other solvents concentration gradient.

Mutual diffusion coefficients were calculated using multicomponent diffusion models. There are several models available to relate self-diffusion coefficient to mutual diffusion coefficient (Alsoy & Duda, 1999; Dabral, 1999; Price & Romdhane, 2003; Zielinski & Hanley, 1999). In this work only Alsoy and Duda model has been used and given in Table 1.

Self diffusion coefficients were calculated using Vrentas and Duda (1977a, 1977b) free volume theory:

$$D_i = D_{0i}\exp\left(-\frac{\sum_{j=1}^{3}\omega_j\hat{V}_j^*(\xi_{i3}/\xi_{j3})}{\hat{V}_{FH}/\gamma}\right) \tag{3}$$

$$\xi_{i3} = \frac{\text{critical molar volume of a jumping unit of component } i}{\text{critical molar volume of the jumping unit of the polymer}}$$

$\xi_{i3} = \frac{\hat{V}_i^* M_{ji}}{\hat{V}_3^* M_{j3}}$ (Vrentas, Duda, & Ling, 1984), and

The hole free volume is given by:

$$\frac{\hat{V}_{FH}}{\gamma} = \frac{K_{11}}{\gamma}\omega_1(K_{21} + T - T_{g1}) + \frac{K_{12}}{\gamma}\omega_2(K_{22} + T - T_{g2}) + \frac{K_{13}}{\gamma}\omega_3(K_{23} + T - T_{g3}) \tag{4}$$

ω_j: mass fraction of component j
$\left(\frac{K_{11}}{\gamma}, K_{21} - T_{g1}\right)$ and $\left(\frac{K_{12}}{\gamma}, K_{22} - T_{g2}\right)$ are free volume parameters for solvent 1 and solvent two respectively
$\frac{K_{13}}{\gamma}, K_{23} - T_{g3}$: are free volume parameters for polymer

Table 1
Four cases for diffusion coefficients of ternary polymer solvent systems (Alsoy & Duda, 1999).

Case	D_{11}	D_{12}	D_{21}	D_{22}
1	$D_1\left[\frac{\partial \ln a_1}{\partial \ln c_1}\right]$	$\frac{c_1}{c_2}D_1\left[\frac{\partial \ln a_1}{\partial \ln c_2}\right]$	$\frac{c_2}{c_1}D_2\left[\frac{\partial \ln a_2}{\partial \ln c_1}\right]$	$D_2\left[\frac{\partial \ln a_2}{\partial \ln c_2}\right]$
2	$D_1\left[\frac{\partial \ln a_1}{\partial \ln c_1}\right]$	0	0	$D_2\left[\frac{\partial \ln a_2}{\partial \ln c_2}\right]$
3	D_1	0	0	D_2
4	$D_1c_1(1 - c_1\hat{V}_1)\left[\frac{\partial \ln a_1}{\partial c_1}\right]$ $-D_2c_1c_2\hat{V}_2\left[\frac{\partial \ln a_2}{\partial c_1}\right]$	$D_1c_1(1 - c_1\hat{V}_1)\left[\frac{\partial \ln a_1}{\partial c_2}\right]$ $-D_2c_1c_2\hat{V}_2\left[\frac{\partial \ln a_2}{\partial c_2}\right]$	$D_2c_2(1 - c_2\hat{V}_2)\left[\frac{\partial \ln a_2}{\partial c_1}\right]$ $-D_1c_1c_2\hat{V}_1\left[\frac{\partial \ln a_1}{\partial c_1}\right]$	$D_2c_2(1 - c_2\hat{V}_2)\left[\frac{\partial \ln a_2}{\partial c_2}\right]$ $-D_1c_1c_2\hat{V}_1\left[\frac{\partial \ln a_1}{\partial c_2}\right]$

2.2. Shrinkage of coating

Coating shrinks due to departure of both the solvents into the room air.

$$\frac{dL}{dt} = -\overline{V}_1 k_1^G (p_{1i}^G - p_{1b}^G) - \overline{V}_2 k_2^G (p_{2i}^G - p_{2b}^G) \tag{5}$$

L is the thickness of coating; k_1^G and k_2^G, are the convective mass transfer coefficients of solvent 1 and solvent 2, respectively; $\overline{V}_1$ and $\overline{V}_2$, are the partial molar volume of solvent 1 and 2, respectively; p_{1b}^G and p_{2b}^G, are partial pressures of solvents 1 and 2 in bulk air, respectively; p_{1i}^G, p_{2i}^G are equilibrium partial pressure of solvent 1 and solvent 2, respectively, and they can be calculated by

$$p_{1i} = P_1^{vap}(T) \cdot \varphi_1 \cdot \gamma_1 \tag{6}$$

$$p_{2i} = P_2^{vap}(T) \cdot \varphi_2 \cdot \gamma_2 \tag{7}$$

φ_i is the volume fraction of species i, P_i^{vap} is saturation vapor pressure of species i, γ_i is the activity constants for species i.

Activity for the ternary systems can be calculated using Flory Huggins theory (Favre, Nguyen, Clement, & Neel, 1996).

Activity coefficient of solvent 1

$$ln\, a_1 = \ln \gamma_1 + \ln \varphi_1 = \ln \varphi_1 + \left(1 - \varphi_1 - \frac{\overline{V}_1}{\overline{V}_2}\varphi_2\right) - \frac{\overline{V}_1}{\overline{V}_3}\varphi_3 + \chi_{13}\varphi_3^2 + \chi_{12}\varphi_2^2 + \varphi_2\varphi_3\left(\chi_{13} + \chi_{12} - \frac{\overline{V}_1}{\overline{V}_2}\chi_{23}\right) \tag{8}$$

Activity coefficient of solvent 2

$$ln\, a_2 = \ln \gamma_2 + \ln \varphi_2 = \ln \varphi_2 + \left(1 - \frac{\overline{V}_2}{\overline{V}_1}\varphi_1 - \varphi_2\right) - \frac{\overline{V}_2}{\overline{V}_3}\varphi_3 + \chi_{23}\varphi_3^2 + \chi_{12}\frac{\overline{V}_2}{\overline{V}_1}\varphi_2^2 + \varphi_1\varphi_3\left(\chi_{12}\frac{\overline{V}_2}{\overline{V}_1} + \chi_{23} - \frac{\overline{V}_2}{\overline{V}_1}\chi_{13}\right) \tag{9}$$

Activity coefficient of polymer 3

$$ln\, a_3 = \ln \varphi_3 + (1 - \varphi_3) - \frac{\bar{V}_3}{\bar{V}_1}\varphi_1 - \frac{\bar{V}_3}{\bar{V}_2}\varphi_2 + \left(\chi_{13}\frac{\bar{V}_3}{\bar{V}_1}\varphi_1 + \chi_{23}\frac{\bar{V}_3}{\bar{V}_2}\varphi_2\right)(\varphi_1 + \varphi_2) - \chi_{12}\frac{\bar{V}_3}{\bar{V}_1}\varphi_1\varphi_2 \tag{10}$$

where χ is the Flory-Huggins binary interaction parameter can be determined from the Bristow and Watson (1958) semi-empirical equation given below,

$$\chi_{ij} = 0.35 + \frac{\overline{V}_i}{RT}(\delta_i - \delta_j)^2 \tag{11}$$

$\overline{V}_i$ is the partial molar volume of solvent i, δ_i is the solubility parameter of solvent i, δ_i is the solubility parameter of polymer j, and volume fraction is given by $\varphi_i = c_i\hat{V}_i$, where c_i is the concentration of species i, $\hat{V}_i$ is the specific volume of species i.

2.3. Energy transport

Usually, the coating is heated by hot air blown on the top and the bottom sides. In the present work, heat transfer occurs due to natural convection only as experiments were made under quiescent conditions. Because coatings are thin, the conductive resistance of the coating is negligible compared to convective resistance in the air. Hence, the coating temperature was assumed to be uniform through the thickness (Alsoy & Duda, 1999). Detailed heat transport model of Price and Cairncross (2000) showed insignificant change in temperature from the top to the bottom of the coating. Also, temperatures of the coating and the substrate were assumed to be same.

The equation for heat transport is given by the following equation:

$$\frac{dT}{dt} = -\left[\frac{h_{top}(T - T^G) + \sum_{i=1}^{N-1} k_{gi}^G \Delta\hat{H}_{vi}(p_{ii}^G - p_{ib}^G) + h_{bottom}(T - T^g)}{\rho^p \hat{C}_p^p X(t) + \rho^s \hat{C}_p^s H}\right] \tag{12}$$

h_{top} and h_{bottom} are the heat transfer coefficients on the top and the bottom sides, respectively. $\Delta\hat{H}_{vi}$ is the enthalpy of evaporation of solvent i, ρ is the density, $\hat{C}_p$ is the specific heat, superscripts, p and s stand for the polymer and the substrate, respectively.

2.4. Boundary conditions at the top surface

Fluxes of the solvents at the top are described by mass transfer coefficients for both the solvents:

Flux of solvent 1:

$$\left.\left(-D_{11}\frac{\partial c_1}{\partial z} - D_{12}\frac{\partial c_2}{\partial z}\right)\right|_{z=L(t)} = (1 - c_1\overline{V}_1)k_1^G(p_{1i}^G - p_{1b}^G) - c_1\overline{V}_2 k_2^G(p_{2i}^G - p_{2b}^G) \tag{13}$$

Flux of solvent 2:

$$\left.\left(-D_{22}\frac{\partial c_2}{\partial z} - D_{21}\frac{\partial c_1}{\partial z}\right)\right|_{z=L(t)} = (1 - c_2\overline{V}_2)k_2^G(p_{2i}^G - p_{2b}^G) - c_2\overline{V}_1 k_2^G(p_{1i}^G - p_{1b}^G) \tag{14}$$

2.5. Boundary conditions at the bottom

Since the substrate is impermeable, fluxes of both the solvents are zero there:
Flux of solvent 1:

$$\left(-D_{11}\frac{\partial c_1}{\partial z}-D_{12}\frac{\partial c_2}{\partial z}\right)\bigg|_{z=0}=0 \tag{15}$$

Flux of solvent 2:

$$\left(-D_{22}\frac{\partial c_2}{\partial z}-D_{21}\frac{\partial c_1}{\partial z}\right)\bigg|_{z=0}=0 \tag{16}$$

3. Solution of equations

Eqs. (1) and (2) are partial differential equations, and Eqs. (5) and (12) are ordinary differential equations; they are coupled and non-linear. Together they model the mass and heat transport during drying. They were solved using Galerkin's method of finite elements, which transforms them into ordinary differential equations (ODEs). ODEs were then integrated with time to determine concentrations as a function of time and distance and temperature as a function of time. The method is described below.

3.1. Galerkin's method of finite elements

In finite element method, the unknown variables, which are concentration here (temperature is not discretized because an ordinary differential equation (Eq. (12)) exists for it), are expressed as a sum of product of unknown coefficients and basis functions:

$$c_1=\sum_{j=1}^{n}u_j\phi_j \tag{17}$$

$$c_2=\sum_{j=1}^{n}v_j\phi_j \tag{18}$$

ϕ_i are basis functions, n is number of nodes in the domain at which the solution is computed and u_j and v_j are unknown coefficients. The basis functions, ϕ_j, are chosen such that they have a value of 1 at node j and 0 at others. This choice of the basis functions renders the computed coefficients, u_j and v_j, as solution at the nodes. The functions are usually piece-wise continuous polynomials of a certain degree because they lend themselves to easy integration (Strang and Fix, 1973). Eqs. (17) and (18) for concentrations are substituted into the governing differential equations (1) and (2) and the residuals are made orthogonal to all functions from a complete set of infinite number of independent functions. For practical purposes, however, finite numbers of functions, as warranted by a trade-off between accuracy and computation time, are taken from the set.

Differentiating equation (17) with respect to t using chain rule,

$$\frac{\partial c_1}{\partial t}=\sum_{j=1}^{n}\frac{\partial u_j}{\partial t}\phi_j+\sum_{j=1}^{n}u_j\frac{\partial \phi_j}{\partial t} \tag{19}$$

Since ϕ_j s are functions of space only and not of time, the second term in Eq. (19) vanishes. Therefore,

$$\frac{\partial c_1}{\partial t}=\sum_{j=1}^{n}\frac{\partial u_j}{\partial t}\phi_j \tag{20}$$

Differentiating equation (17) with respect to z,

$$\frac{\partial c_1}{\partial z}=\sum_{j=1}^{n}\frac{\partial u_j}{\partial z}\phi_j+\sum_{j=1}^{n}u_j\frac{\partial \phi_j}{\partial z} \tag{21}$$

Since u_j are unknown coefficients at j, the first term vanishes in Eq. (21). Hence,

$$\frac{\partial c_1}{\partial z}=\sum_{j=1}^{n}u_j\frac{\partial \phi_j}{\partial z} \tag{22}$$

Similarly,

$$\frac{\partial c_2}{\partial t}=\sum_{j=1}^{n}\frac{\partial v_j}{\partial t}\phi_j \tag{23}$$

and

$$\frac{\partial c_2}{\partial z}=\sum_{j=1}^{n} v_j\frac{\partial \phi_j}{\partial z} \tag{24}$$

Residual of Eqs. (1) and (2), R_1 and R_2, are made orthogonal to each function, ψ, belonging to a complete set of functions.

$$R_1=\int_0^L\left[\frac{\partial c_1}{\partial t}-\frac{\partial}{\partial z}\left(D_{11}\frac{\partial c_1}{\partial z}\right)-\frac{\partial}{\partial z}\left(D_{12}\frac{\partial c_2}{\partial z}\right)\right]\psi_i\,dz,\quad i=1,2,3,\ldots,n \tag{25}$$

$$R_2=\int_0^L\left[\frac{\partial c_2}{\partial t}-\frac{\partial}{\partial z}\left(D_{21}\frac{\partial c_1}{\partial z}\right)-\frac{\partial}{\partial z}\left(D_{22}\frac{\partial c_2}{\partial z}\right)\right]\psi_i\,dz,\quad i=1,2,3,\ldots,n \tag{26}$$

Substituting the expressions for $\partial c_1/\partial t$, $\partial c_1/\partial z$ and $\partial c_2/\partial z$ from Eqs. (21), (22) and (24) in Eq. (25),

$$\begin{aligned}R_1&=\int_0^L\left[\sum_{j=1}^{n}\frac{\partial u_j}{\partial t}\phi_j-\frac{\partial}{\partial z}\left(D_{11}\sum_{j=1}^{n}u_j\frac{\partial \phi_j}{\partial z}\right)-\frac{\partial}{\partial z}\left(D_{12}\sum_{j=1}^{n}v_j\frac{\partial \phi_j}{\partial z}\right)\right]\psi_i\,dz\\ R_1&=\int_0^L\left(\sum_{j=1}^{n}\frac{\partial u_j}{\partial t}\phi_j\right)\psi_i dz-\int_0^L\left[\frac{\partial}{\partial z}\left(D_{11}\sum_{j=1}^{n}u_j\frac{\partial \phi_j}{\partial z}\right)\right]\psi_i dz-\int_0^L\left[\frac{\partial}{\partial z}\left(D_{12}\sum_{j=1}^{n}v_j\frac{\partial \phi_j}{\partial z}\right)\right]\psi_i\,dz\end{aligned} \tag{27}$$

Substituting the expressions for $\partial c_1/\partial t$, $\partial c_2/\partial z$ and $\partial c_2/\partial z$ from Eqs. (22)–(24) in Eq. (26),

$$R_2=\int_0^L\left[\sum_{j=1}^{n}\frac{\partial v_j}{\partial t}\phi_j\right]\psi_i dz-\int_0^L\left[\frac{\partial}{\partial z}\left(D_{21}\sum_{j=1}^{n}u_j\frac{\partial \phi_j}{\partial z}\right)\right]\psi_i dz-\int_0^L\left[\frac{\partial}{\partial z}\left(D_{22}\sum_{j=1}^{n}v_j\frac{\partial \phi_j}{\partial z}\right)\right]\psi_i dz \tag{28}$$

The integration and summation in Term 1 of Eq. (27) can interchanged without any loss of generality. Hence, Term 1 becomes

$$\int_0^L\left[\sum_{j=1}^{n}\frac{\partial u_j}{\partial t}\varphi_j\right]\psi_i\,dz=\int_0^L\left[\sum_{j=1}^{n}\phi_j\psi_i\,dz\right]\frac{\partial u_j}{\partial t}=\sum_{j=1}^{n}\left(\int_0^L\phi_j\psi_i dz\right)\frac{\partial u_j}{\partial t},\quad i=1,2,3,\ldots,n,\quad j=1,2,3,\ldots,n \tag{29}$$

Term 1 of Eq. (27) can be expanded for all values of i and j and, arranged in matrix form:

$$\begin{bmatrix}\int_0^L\phi_1\psi_1 dz & \int_0^L\phi_2\psi_1 dz & \cdots & \int_0^L\phi_n\psi_1 dz\\ \vdots & \vdots & \cdots & \vdots\\ \int_0^L\phi_1\psi_n dz & \int_0^L\phi_2\psi_n dz & \cdots & \int_0^L\phi_n\psi_n dz\end{bmatrix}_{n\times n}\begin{bmatrix}\frac{\partial u_1}{\partial t}\\ \vdots\\ \frac{\partial u_n}{\partial t}\end{bmatrix}_{n\times 1}=\boldsymbol{A_1}\dot{\boldsymbol{U}} \tag{30}$$

Terms 2 and 3 of Eq. (27) can be integrated by parts to give

$$\begin{aligned}&\int_0^L\left\{\frac{\partial}{\partial z}\left(D_{11}\sum_{j=1}^{n}\frac{\partial \phi_j}{\partial z}u_j\right)\right\}\psi_i\,dz+\int_0^L\left\{\frac{\partial}{\partial z}\left(D_{12}\sum_{j=1}^{n}\frac{\partial \phi_j}{\partial z}v_j\right)\right\}\psi_i\,dz=\left[D_{11}\sum_{j=1}^{n}\frac{\partial \phi_j}{\partial z}u_j\psi_i\Big|_0^L-\int_0^L\left(D_{11}\sum_{j=1}^{n}\frac{\partial \phi_j}{\partial z}\frac{\partial \psi_i}{\partial z}u_j\right)dz\right.\\ &\left.+D_{12}\sum_{j=1}^{n}\frac{\partial \phi_j}{\partial z}v_j\psi_i\Big|_0^L-\int_0^L\left(D_{12}\sum_{j=1}^{n}\frac{\partial \phi_j}{\partial z}\frac{\partial \psi_i}{\partial z}v_j\right)dz\right]=\left[D_{11}\sum_{j=1}^{n}\frac{\partial \phi_j}{\partial z}u_j\psi_i\Big|_0^L+D_{12}\sum_{j=1}^{n}\frac{\partial \phi_j}{\partial z}v_j\psi_i\Big|_0^L\cdots-\int_0^L\left(D_{11}\sum_{j=1}^{n}\frac{\partial \phi_j}{\partial z}\frac{\partial \psi_i}{\partial z}u_j\right)dz\right.\\ &\left.-\int_0^L\left(D_{12}\sum_{j=1}^{n}\frac{\partial \phi_j}{\partial z}\frac{\partial \psi_i}{\partial z}v_j\right)dz\right]\end{aligned} \tag{31}$$

The first and second terms of Eq. (31) are evaluated at $z=0$ and $z=L$. They constitute two boundary conditions for solvent 1.
At the coating-air interface, $z=L$, only basis function ψ_1 has a value of one and others are zero. Therefore, boundary condition there is:

$$\begin{aligned}&D_{11}\left[\sum_{j=1}^{n}\frac{\partial \phi_j}{\partial z}u_j\right]\psi_1+D_{12}\left[\sum_{j=1}^{n}\frac{\partial \phi_j}{\partial z}v_j\right]\psi_1=D_{11}\left[\frac{\partial}{\partial z}\left(\sum_{j=1}^{n}u_j\phi_j\right)\right]+D_{12}\left[\frac{\partial}{\partial z}\left(\sum_{j=1}^{n}v_j\phi_j\right)\right],\\ &(\because\psi_1=1,\quad\text{at top})=D_{11}\left[\frac{\partial c_1}{\partial z}\right]+D_{12}\left[\frac{\partial c_2}{\partial z}\right]=\left[D_{11}\frac{\partial c_1}{\partial z}+D_{12}\frac{\partial c_2}{\partial z}\right]=-[(1-c_1\overline{V}_1)k_{g1}(p_{1i}-p_{1b})-c_1\overline{V}_2k_{g2}(p_{2i}-p_{2b})]\end{aligned} \tag{32}$$

Eq. (32) indicates that the boundary condition at the top is described by mass transfer coefficients for solvents 1 and 2.

At the coating-substrate interface, ψ_n has a value of one and others are zero. Hence,

$$D_{11}\left[\sum_{j=1}^{n}\frac{\partial\phi_j}{\partial z}u_j\right]\psi_n + D_{12}\left[\sum_{j=1}^{n}\frac{\partial\phi_j}{\partial z}v_j\right]\psi_n = D_{11}\left[\frac{\partial}{\partial z}\left(\sum_{j=1}^{n}u_j\phi_j\right)\right]\psi_n + D_{12}\left[\frac{\partial}{\partial z}\left(\sum_{j=1}^{n}v_j\phi_j\right)\right]\psi_n,$$

$$(\psi_n = 1,\quad \text{at bottom}) = D_{11}\left[\frac{\partial c_1}{\partial z}\right]\psi_n + D_{12}\left[\frac{\partial c_2}{\partial z}\right]\psi_n == \left[D_{11}\frac{\partial c_1}{\partial z} + D_{12}\frac{\partial c_2}{\partial z}\right]\psi_n = 0 \tag{33}$$

Eq. (33) means that the flux of the solvent at the coating-substrate interface is zero.

The boundary conditions were arranged in the form of a vector C_1 as:

$$C_1 = \begin{bmatrix} -[(1-c_1\overline{V}_1)k_{g1}(p_{1i}-p_{1b}) - c_1\overline{V}_2k_{g2}(p_{2i}-p_{2b})] & \leftarrow \text{Boundary condition at top} \\ 0 & \\ \vdots & \\ 0 & \leftarrow \text{Boundary condition at bottom} \end{bmatrix}_{n\times 1} \tag{34}$$

Without loss of generality, integration and summation can be interchanged in third and fourth term of Eq. (31). Therefore,

$$\int_0^L\left(D_{11}\sum_{j=1}^{n}\frac{\partial\phi_j}{\partial z}\frac{\partial\psi_i}{\partial z}u_j\right)dz + \int_0^L\left(D_{12}\sum_{j=1}^{n}\frac{\partial\phi_j}{\partial z}\frac{\partial\psi_i}{\partial z}v_j\right)dz = \sum_{j=1}^{n}\left(\int_0^L D_{11}\frac{\partial\phi_j}{\partial z}\frac{\partial\psi_i}{\partial z}dz\right)u_j + \sum_{j=1}^{n}\left(\int_0^L D_{12}\frac{\partial\phi_j}{\partial z}\frac{\partial\psi_i}{\partial z}dz\right)v_j,$$

$$i = 1, 2, 3, \ldots, n \tag{35}$$

Eq. (35) can be expanded for all values of i and j and rearranged in matrix form as:

$$\begin{bmatrix} \int_0^L D_{11}\frac{d\phi_1}{dz}\frac{d\psi_1}{dz}dz & \cdots & \int_0^L D_{11}\frac{d\phi_n}{dz}\frac{d\psi_1}{dz}dz & \int_0^L D_{12}\frac{d\phi_1}{dz}\frac{d\psi_1}{dz}dz & \cdots & \int_0^L D_{12}\frac{d\phi_n}{dz}\frac{d\psi_1}{dz}dz \\ \int_0^L D_{11}\frac{d\phi_1}{dz}\frac{d\psi_2}{dz}dz & \cdots & \int_0^L D_{11}\frac{d\phi_n}{dz}\frac{d\psi_2}{dz}dz & \int_0^L D_{12}\frac{d\phi_1}{dz}\frac{d\psi_2}{dz}dz & \cdots & \int_0^L D_{12}\frac{d\phi_n}{dz}\frac{d\psi_2}{dz}dz \\ \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ \int_0^L D_{11}\frac{d\phi_1}{dz}\frac{d\psi_n}{dz}dz & \cdots & \int_0^L D_{11}\frac{d\phi_n}{dz}\frac{d\psi_n}{dz}dz & \int_0^L D_{12}\frac{d\phi_1}{dz}\frac{d\psi_n}{dz}dz & \cdots & \int_0^L D_{12}\frac{d\phi_n}{dz}\frac{d\psi_n}{dz}dz \end{bmatrix}_{n\times 2n} \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \\ v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix}_{2n\times 1} = \boldsymbol{B}_1\boldsymbol{U} \tag{36}$$

Now substituting Terms 1–3 into Eq. (27)

$$R_1 = A_1\dot{U}_1 - B_1U - C_1 \tag{37}$$

Similar to Eq. (37) for residual R_1, the following matrix equation can be written for residual R_2 (Eq. (28)):

$$R_2 = A_2\dot{V}_2 - B_2U_2 - C_2 = 0 \tag{38}$$

$$\begin{bmatrix} \int_0^L \phi_1\psi_1dz & \int_0^L \phi_2\psi_1dz & \cdots & \int_0^L \phi_n\psi_1dz \\ \vdots & \vdots & \cdots & \vdots \\ \int_0^L \phi_1\psi_ndz & \int_0^L \phi_2\psi_ndz & \cdots & \int_0^L \phi_n\psi_ndz \end{bmatrix}_{n\times n} \begin{bmatrix} \frac{\partial v_1}{\partial t} \\ \vdots \\ \frac{\partial v_n}{\partial t} \end{bmatrix}_{n\times 1} = \boldsymbol{A}_2\dot{\boldsymbol{V}} \tag{39}$$

$$C_2 = \begin{bmatrix} -[(1-c_2\overline{V_2})k_{g2}(p_{2i}-p_{2b}) - c_2\overline{V_2}k_{g1}(p_{1i}-p_{1b})] & \leftarrow \text{Boundary condition at top} \\ 0 & \\ \vdots & \\ 0 & \leftarrow \text{Boundary condition at bottom} \end{bmatrix} \tag{40}$$

Eq. (28) can be expanded in matrix form similar to Eq. (27)

$$B_2U_2=\begin{bmatrix}\int_0^L D_{22}\frac{d\phi_1}{dz}\frac{d\psi_1}{dz}dz & \cdots & \int_0^L D_{22}\frac{d\phi_n}{dz}\frac{d\psi_1}{dz}dz & \int_0^L D_{21}\frac{d\phi_1}{dz}\frac{d\psi_1}{dz}dz & \cdots & \int_0^L D_{21}\frac{d\phi_n}{dz}\frac{d\psi_1}{dz}dz\\ \int_0^L D_{22}\frac{d\phi_1}{dz}\frac{d\psi_2}{dz}dz & \cdots & \int_0^L D_{22}\frac{d\phi_n}{dz}\frac{d\psi_2}{dz}dz & \int_0^L D_{21}\frac{d\phi_1}{dz}\frac{d\psi_2}{dz}dz & \cdots & \int_0^L D_{21}\frac{d\phi_n}{dz}\frac{d\psi_2}{dz}dz\\ \vdots & \cdots & \vdots & \vdots & \cdots & \vdots\\ \int_0^L D_{22}\frac{d\phi_1}{dz}\frac{d\psi_n}{dz}dz & \cdots & \int_0^L D_{22}\frac{d\phi_n}{dz}\frac{d\psi_n}{dz}dz & \int_0^L D_{21}\frac{d\phi_1}{dz}\frac{d\psi_n}{dz}dz & \cdots & \int_0^L D_{21}\frac{d\phi_n}{dz}\frac{d\psi_n}{dz}dz\end{bmatrix}_{n*2n}\begin{bmatrix}v_1\\ v_2\\ \vdots\\ v_n\\ u_1\\ u_2\\ \vdots\\ u_n\end{bmatrix}_{2n\times 1}$$

Above matrix can be rearranged as

$$\begin{bmatrix}\int_0^L D_{21}\frac{d\phi_1}{dz}\frac{d\psi_1}{dz}dz & \cdots & \int_0^L D_{21}\frac{d\phi_n}{dz}\frac{d\psi_1}{dz}dz & \int_0^L D_{22}\frac{d\phi_1}{dz}\frac{d\psi_1}{dz}dz & \cdots & \int_0^L D_{22}\frac{d\phi_n}{dz}\frac{d\psi_1}{dz}dz\\ \int_0^L D_{21}\frac{d\phi_1}{dz}\frac{d\psi_2}{dz}dz & \cdots & \int_0^L D_{21}\frac{d\phi_n}{dz}\frac{d\psi_2}{dz}dz & \int_0^L D_{22}\frac{d\phi_1}{dz}\frac{d\psi_2}{dz}dz & \cdots & \int_0^L D_{22}\frac{d\phi_n}{dz}\frac{d\psi_2}{dz}dz\\ \vdots & \cdots & \vdots & \vdots & \cdots & \vdots\\ \int_0^L D_{21}\frac{d\phi_1}{dz}\frac{d\psi_n}{dz}dz & \cdots & \int_0^L D_{21}\frac{d\phi_n}{dz}\frac{d\psi_n}{dz}dz & \int_0^L D_{22}\frac{d\phi_1}{dz}\frac{d\psi_n}{dz}dz & \cdots & \int_0^L D_{22}\frac{d\phi_n}{dz}\frac{d\psi_n}{dz}dz\end{bmatrix}_{n\times 2n}\begin{bmatrix}u_1\\ u_2\\ \vdots\\ u_n\\ v_1\\ v_2\\ \vdots\\ v_n\end{bmatrix}_{2n\times 1}=\boldsymbol{B_2U}\tag{41}$$

Matrix Eqs. (37) and (38) can be combined and arranged as:

$$\boldsymbol{A\dot{U}}=\boldsymbol{BU}+\boldsymbol{C}\tag{42}$$

$$\boldsymbol{A}=\begin{bmatrix}\int_0^L \phi_1\psi_1dz & \cdots & \int_0^L \phi_n\psi_1dz & \int_0^L \phi_{n+1}\psi_1dz & & \int_0^L \phi_{2n}\psi_1dz\\ \vdots & \cdots & \vdots & \vdots & \cdots & \vdots\\ \int_0^L \phi_1\psi_ndz & \cdots & \int_0^L \phi_n\psi_ndz & \int_0^L \phi_{n+1}\psi_ndz & \cdots & \int_0^L \phi_{2n}\psi_ndz\\ \int_0^L \phi_1\psi_{n+1}dz & \cdots & \int_0^L \phi_n\psi_{n+1}dz & \int_0^L \phi_{n+1}\psi_{n+1}dz & \cdots & \int_0^L \phi_{2n}\psi_{n+1}dz\\ \vdots & \cdots & \vdots & \vdots & \cdots & \vdots\\ \int\limits_0^L \phi_1\psi_{2n}dz & \cdots & \int\limits_0^L \phi_n\psi_{2n}dz & \int\limits_0^L \phi_1\psi_{2n}dz & \cdots & \int_0^L \phi_n\psi_{2n}dz\end{bmatrix}_{2n\times 2n}\qquad U=\begin{bmatrix}u_1\\ u_2\\ \vdots\\ \vdots\\ u_n\\ v_1\\ v_2\\ \vdots\\ \vdots\\ v_n\end{bmatrix}_{2n\times 1}$$

$$\boldsymbol{\dot{U}}=\begin{bmatrix}\frac{\partial u_1}{\partial t}\\ \vdots\\ \frac{\partial u_n}{\partial t}\\ \frac{\partial v_1}{\partial t}\\ \vdots\\ \frac{\partial v_n}{\partial t}\end{bmatrix}_{2n\times 1}\qquad \boldsymbol{C}=\begin{bmatrix}-\left[(1-c_1\overline{V}_1)k_{g1}(p_{1i}-p_{1b})-c_1\bar{V}_2k_{g2}(p_{2i}-p_{2b})\right]\\ 0\\ \vdots\\ 0\\ -[(1-c_2\overline{V}_2)k_{g2}(p_{2i}-p_{2b})-c_2\overline{V}_1k_{g1}(p_{1i}-p_{1b})]\\ 0\\ \vdots\\ 0\end{bmatrix}_{2n\times 1}$$

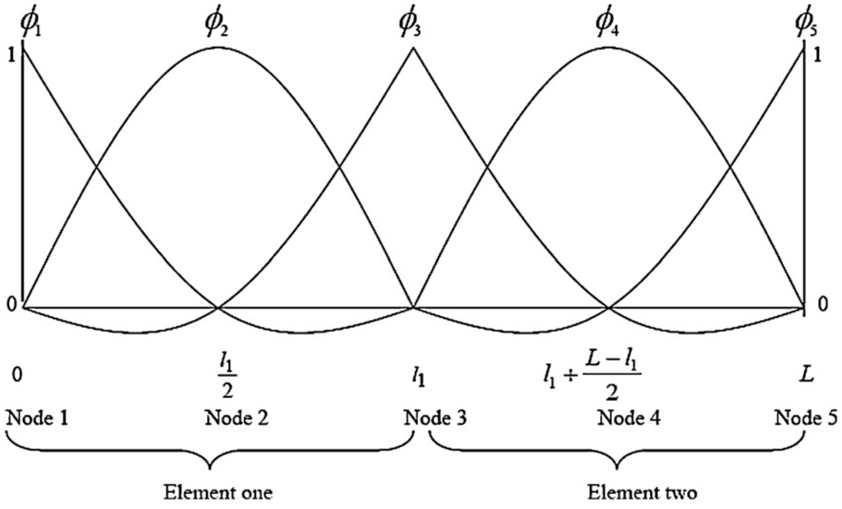


Fig. 2. Quadratic basis functions for two elements. Each element has three nodes and the same number of basis functions. The node within an element is chosen to be at its center.

$$\boldsymbol{B}=\begin{bmatrix}
\int_0^L D_{11}\frac{d\phi_1}{dz}\frac{d\psi_1}{dz}dz & \cdots & \int_0^L D_{11}\frac{d\phi_n}{dz}\frac{d\psi_1}{dz}dz & \int_0^L D_{12}\frac{d\phi_1}{dz}\frac{d\psi_1}{dz}dz & \cdots & \int_0^L D_{12}\frac{d\phi_n}{dz}\frac{d\psi_1}{dz}dz \\
\vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\
\int_0^L D_{11}\frac{d\phi_1}{dz}\frac{d\psi_n}{dz}dz & \cdots & \int_0^L D_{11}\frac{d\phi_n}{dz}\frac{d\psi_n}{dz}dz & \int_0^L D_{12}\frac{d\phi_1}{dz}\frac{d\psi_n}{dz}dz & \cdots & \int_0^L D_{12}\frac{d\phi_n}{dz}\frac{d\psi_n}{dz}dz \\
\int_0^L D_{21}\frac{d\phi_1}{dz}\frac{d\psi_1}{dz}dz & \cdots & \int_0^L D_{11}\frac{d\phi_n}{dz}\frac{d\psi_1}{dz}dz & \int_0^L D_{22}\frac{d\phi_1}{dz}\frac{d\psi_1}{dz}dz & \cdots & \int_0^L D_{22}\frac{d\phi_n}{dz}\frac{d\psi_1}{dz}dz \\
\vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\
\int_0^L D_{21}\frac{d\phi_1}{dz}\frac{d\psi_n}{dz}dz & \cdots & \int_0^L D_{21}\frac{d\phi_n}{dz}\frac{d\psi_n}{dz}dz & \int_0^L D_{22}\frac{d\phi_1}{dz}\frac{d\psi_n}{dz}dz & \cdots & \int_0^L D_{22}\frac{d\phi_n}{dz}\frac{d\psi_n}{dz}dz
\end{bmatrix}_{2n\times 2n}$$

3.2. Evaluation of matrices *A* and *B*

In Galerkin's method, the functions, ψ_j are chosen to be same as the basis functions, ϕ_j. In this work, the basis functions were chosen to be quadratic polynomials because of their popularity. For quadratic polynomials, each element has three nodes—one at each end and the third one at the center as shown in Fig. 2. As mentioned before, the polynomials chosen were piece-wise continuous having a value of 1 at one node and zero at all other nodes. For example, the polynomial $\phi_1=1$ at $z=0$ and $\phi_1=0$ at $z=l_1/2$ and $z=l_1$. Each polynomial spans at the most two elements. For illustrative purpose, only two elements are shown in Fig. 2. The number of elements can be changed till the solution becomes independent of it.

Equations for a basis functions are given below for an element starting at a and ending at b.

$$\phi_1=\frac{2}{(a-b)^2}z^2-\frac{a+3b}{(a-b)^2}z+\frac{b^2+ab}{(a-b)^2} \tag{43}$$

$$\phi_2=-\frac{4}{(a-b)^2}z^2+4\frac{a+b}{(a-b)^2}z-\frac{4ab}{(a-b)^2} \tag{44}$$

$$\phi_3=\frac{2}{(a-b)^2}z^2-\frac{3a+b}{(a-b)^2}z+\frac{a^2+ab}{(a-b)^2} \tag{45}$$

It can easily be seen that the basis functions ϕ_i ($i=1$, 2 and 3) have a value of one at node i and zero at others. In Fig. 2, for the first element, $a=0$ and $b=1$ and for the second, $a=l_1$ and $b=L$. Unlike other basis functions, ϕ_3, spans two elements. Eq. (45) defines it in element one and in element two it is same as ϕ_1 in element one for different values of a and b.

Now, substituting the values of a and b in elements one and two in Eqs. (43)–(45), the basis functions in Fig. 2 are obtained:

3.3. First element

$$\phi_1=\frac{2}{{l_1}^2}z^2-\frac{3b}{{l_1}^2}z+1 \tag{46}$$

$$\phi_2 = -\frac{4}{{l_1}^2}z^2 + \frac{4}{l_1}z \tag{47}$$

$$\phi_3 = \frac{2}{{l_1}^2}z^2 - \frac{1}{l_1}z + \frac{1}{l_1} \tag{48}$$

3.4. Second element

$$\phi_3 = \frac{2}{(l_1-L)^2}z^2 - \frac{l_1+3L}{(l_1-L)^2}z + \frac{L^2+l_1L}{(l_1-L)^2} \tag{49}$$

$$\phi_4 = -\frac{4}{(l_1-L)^2}z^2 + 4\frac{l_1+L}{(l_1-L)^2}z - \frac{4l_1L}{(l_1-L)^2} \tag{50}$$

$$\phi_5 = \frac{2}{(l_1-L)^2}z^2 - \frac{3l_1+L}{(l_1-L)^2}z + \frac{{l_1}^2+l_1L}{(l_1-L)^2} \tag{51}$$

In Galerkin's method, the weighting functions ψ_i are chosen to be same as the basis functions, ϕ_i. Incorporating $\psi_i=\phi_i$ in matrix $\boldsymbol{A}$ changes it to:

$$\boldsymbol{A} = \begin{bmatrix} \int_0^L \phi_1\phi_1 dz & \cdots & \int_0^L \phi_n\phi_1 dz & \int_0^L \phi_{n+1}\phi_1 dz & \cdots & \int_0^L \phi_{2n}\phi_1 dz \\ \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ \int_0^L \phi_1\phi_n dz & \cdots & \int_0^L \phi_n\phi_n dz & \int_0^L \phi_{n+1}\phi_n dz & \cdots & \int_0^L \phi_{2n}\phi_n dz \\ \int_0^L \phi_1\phi_{n+1} dz & \cdots & \int_0^L \phi_n\phi_{n+1} dz & \int_0^L \phi_{n+1}\phi_{n+1} dz & \cdots & \int_0^L \phi_{2n}\phi_{n+1} dz \\ \vdots & \cdots & \vdots & \vdots & \cdots & \vdots \\ \int_0^L \phi_1\phi_{2n} dz & \cdots & \int_0^L \phi_n\phi_{2n} dz & \int_0^L \phi_{n+1}\phi_{2n} dz & \cdots & \int_0^L \phi_{2n}\phi_{2n} dz \end{bmatrix}_{2n\times 2n}$$

The evaluation of entries in row 1 is shown below for two elements. Entries in row 1 and entries in others rows for any number of elements can be calculated in a similar way.

$A_{11}=\int_0^L \phi_1\phi_1 = \int_0^{l_1}\phi_1\phi_1 + \int_{l_1}^L \phi_1\phi_1 = \int_0^{l_1}\phi_1\phi_1$, because ϕ_1 is zero in element two.

For the same reason as above, the next two entries in row one are:

$$A_{12} = \int_0^L \phi_2\phi_1 = \int_0^{l_1}\phi_2\phi_1 + \int_{l_1}^L \phi_2\phi_1 = \int_0^{l_1}\phi_2\phi_1$$

$$A_{13} = \int_0^L \phi_3\phi_1 = \int_0^{l_1}\phi_3\phi_1 + \int_{l_1}^L \phi_3\phi_1 = \int_0^{l_1}\phi_3\phi_1$$

The remaining entries in row one are zeros because in element one ϕ_4 and ϕ_5 are zero and in element two ϕ_i is zero.

$$A_{14} = \int_0^L \phi_4\phi_1 = \int_0^{l_1}\phi_4\phi_1 + \int_{l_1}^L \phi_4\phi_1 = 0$$

$$A_{15} = \int_0^L \phi_5\phi_1 = \int_0^{l_1}\phi_5\phi_1 + \int_{l_1}^L \phi_5\phi_1 = 0$$

Similar calculation can be shown for solvent 2.

$$A_{n+1,1}=A_{11},\quad A_{n+1,2}=A_{12},\quad A_{n+1,3}=A_{13},\quad A_{n+1,4}=A_{14},\quad A_{n+1,5}=A_{15}$$

The second row would also have only six non-zero entries because ϕ_4 and ϕ_5 are zeros in element one. The third row would have ten non-zero entries because the basis function ϕ_3 spans both the elements.

For n elements, starting with row three, every odd row would have ten non-zero entries and every even row would have six non-zero entries. That is, 4th, 6th, 8th,. . . would have six non-zero entries and 5th, 7th, 9th,. . . would have ten non-zero entries. The matrix $\boldsymbol{A}$ then has the following form with A_{ij} $(i, j = 1, 2, \ldots, n)$ representing non-zero entries.

$$A = \begin{bmatrix}
A_{11} & A_{12} & A_{13} & 0 & 0 & 0 & 0 & A_{11} & A_{12} & A_{13} & 0 & 0 & 0 & 0 \\
A_{21} & A_{22} & A_{23} & 0 & 0 & 0 & 0 & A_{21} & A_{22} & A_{23} & 0 & 0 & 0 & 0 \\
A_{31} & A_{32} & A_{33} & A_{34} & A_{35} & 0 & 0 & A_{31} & A_{32} & A_{33} & A_{34} & A_{35} & 0 & 0 \\
0 & 0 & A_{43} & A_{44} & A_{45} & 0 & 0 & 0 & 0 & A_{43} & A_{44} & A_{45} & 0 & 0 \\
0 & 0 & A_{53} & A_{54} & A_{55} & A_{56} & A_{57} & 0 & 0 & A_{53} & A_{54} & A_{55} & A_{56} & A_{57} \\
0 & 0 & 0 & 0 & A_{65} & A_{66} & A_{67} & 0 & 0 & 0 & 0 & A_{65} & A_{66} & A_{67} \\
0 & 0 & 0 & 0 & A_{75} & A_{76} & A_{77} & 0 & 0 & 0 & 0 & A_{75} & A_{76} & A_{77} \\
A_{11} & A_{12} & A_{13} & 0 & 0 & 0 & 0 & A_{11} & A_{12} & A_{13} & 0 & 0 & 0 & 0 \\
A_{21} & A_{22} & A_{23} & 0 & 0 & 0 & 0 & A_{21} & A_{22} & A_{23} & 0 & 0 & 0 & 0 \\
A_{31} & A_{32} & A_{33} & A_{34} & A_{35} & 0 & 0 & A_{31} & A_{32} & A_{33} & A_{34} & A_{35} & 0 & 0 \\
0 & 0 & A_{43} & A_{44} & A_{45} & 0 & 0 & 0 & 0 & A_{43} & A_{44} & A_{45} & 0 & 0 \\
0 & 0 & A_{53} & A_{54} & A_{55} & A_{56} & A_{57} & 0 & 0 & A_{53} & A_{54} & A_{55} & A_{56} & A_{57} \\
0 & 0 & 0 & 0 & A_{65} & A_{66} & A_{67} & 0 & 0 & 0 & 0 & A_{65} & A_{66} & A_{67} \\
0 & 0 & 0 & 0 & A_{75} & A_{76} & A_{77} & 0 & 0 & 0 & 0 & A_{75} & A_{76} & A_{77}
\end{bmatrix}_{2n\times 2n}$$

Similar calculations can be made for each row of matrix $\boldsymbol{B}$, which will have the same structure as matrix $\boldsymbol{A}$. The calculation of entries of matrix $\boldsymbol{B}$ is complicated than those in matrix $\boldsymbol{A}$ because of diffusion coefficient. The entries in the matrices are obtained by evaluating the integrals by Gaussian quadrature. To perform repeated evaluations using a computer, it is better to scale all the elements to [−1 1]. After scaling, the entries in matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ would change; only entries in row one of the matrices are illustrated below. Other entries in the matrices change in a similar way.

$$\int_0^{l_1} \phi_1\phi_1\,dz == \frac{l_1}{2}\int_{-1}^{1} \Phi_1\Phi_1 dx, \quad \int_0^{l_1} \phi_2\phi_1 = \frac{l_1}{2}\int_{-1}^{1} \Phi_2\Phi_1 dx, \quad \int_0^{l_1} \phi_3\phi_1 = \frac{l_1}{2}\int_{-1}^{1} \Phi_3\Phi_1 dx \tag{52}$$

where

$$\Phi_1 = \frac{x^2-x}{2}, \quad \Phi_2 = -x^2+1, \quad \Phi_3 = \frac{x^2+x}{2} \tag{53}$$

The entries in first row of matrix $\boldsymbol{B}$ would become:

$$\int_0^{l_1} D\frac{d\phi_1}{dz}\frac{d\phi_1}{dz}dz = \frac{2}{l_1}\int_{-1}^{1} D\frac{d\Phi_1}{dx}\frac{d\Phi_1}{dx}dx \tag{54}$$

$$\int_0^{l_1} D\frac{d\phi_2}{dz}\frac{d\phi_1}{dz}dz = \frac{2}{l_1}\int_{-1}^{1} D\frac{d\Phi_2}{dx}\frac{d\Phi_1}{dx}dx \tag{55}$$

$$\int_0^{l_1} D\frac{d\phi_3}{dz}\frac{d\phi_1}{dz}dz = \frac{2}{l_1}\int_{-1}^{1} D\frac{d\Phi_3}{dx}\frac{d\Phi_1}{dx}dx \tag{56}$$

Mutual diffusion coefficient in matrix $\boldsymbol{B}$ is to be calculated using multicomponent diffusion models given in Table 1 which need value of self diffusion coefficients. Self-diffusion coefficients for both the solvents were calculated at Gauss points. The method is given below.

Calculate the concentration of each solvent at each Gauss points

Gauss points are, $k = \left[\sqrt{15}/5, 0, -\sqrt{15}/5\right]$ corresponding weights, $\varpi = [5/9, 8/9, 5/9]$

$$c_1(k) = \sum_{i=1}^{3} u_i\Phi_i(k) \tag{57}$$

$$c_2(k) = \sum_{i=1}^{3} v_i\Phi_i(k) \tag{58}$$

3.5. Weight fractions at Gauss points

Weight fraction of solvent 1

$$\omega_1(k) = \frac{c_1(k)}{c_1(k)+c_2(k)+(1-c_1(k)\hat{V}_1-c_2(k)\hat{V}_2)\rho_{polymer}} \tag{59}$$

Table 2
Free volume parameters (Alsoy & Duda, 1999).

Parameter	Unit	PS–toluene	PS–tetrahydrofuran
D_0	cm^2/s	4.82×10^{-4}	14.4×10^{-4}
$\frac{K_{11}}{\gamma}$	cm^3/g K	0.000145	0.00075
$\frac{K_{12}}{\gamma}$	cm^3/g K	0.000582	0.000582
K_{21}	K	−86.32	10.45
K_{22}	K	−327	−327
T_{g1}	K	0	0
T_{g2}	K	0	0
$\hat{V}_1^*$	cm^3/g	0.917	0.899
$\hat{V}_2^*$	cm^3/g	0.85	0.85
ξ		0.58	0.45
χ		0.354	0.34

Weight fraction of solvent 2

$$\omega_2(k)=\frac{c_2(k)}{c_1(k)+c_2(k)+(1-c_1(k)\hat{V}_1-c_2(k)\hat{V}_2)\rho_{polymer}} \tag{60}$$

Weight fraction of polymer

$$\omega_3(k)=1-\omega_1(k)-\omega_2(k) \tag{61}$$

Total free volume for jumping at Gauss points

$$\frac{\hat{V}_{FH}}{\gamma}(k)=\frac{K_{11}}{\gamma}\omega_1(k)(K_{21}+T-T_{g1})+\frac{K_{12}}{\gamma}\omega_2(k)(K_{22}+T-T_{g2})+\frac{K_{13}}{\gamma}\omega_3(k)(K_{23}+T-T_{g3}) \tag{62}$$

Now self-diffusion coefficient were calculated using following

$$D_i(k)=D_{0i}\,\exp\left(-\frac{\left(\sum_{j=1}^{3}\omega_j(K)\hat{V}_j^*\xi_{i3}/\xi_{j3}\right)}{\hat{V}_{FH}/\gamma(k)}\right) \tag{63}$$

Above self-diffusion coefficient were used to calculate the mutual diffusion coefficient and then entries in ***B*** matrix were calculated using three point Gauss quadrature, as shown below

Lets take Eq. (54)

$$\int_0^{l_1}D\frac{d\phi_1}{dz}\frac{d\phi_1}{dz}dz=\frac{2}{l_1}\int_{-1}^{1}D\frac{d\Phi_1}{dx}\frac{d\Phi_1}{dx}dx=\frac{2}{l_1}\sum_{k=1}^{3}(D_{11}(k)\cdot\Phi_1'(k)\cdot\Phi_1'(k)\cdot\varpi(k)) \tag{64}$$

Entries in ***A*** matrix were calculated as following

$$\frac{l_1}{2}\int_{-1}^{1}\Phi_1\Phi_1dx==\frac{l_1}{2}\sum_{k=1}^{3}(\Phi_1(k)\cdot\Phi_1(k)\cdot\varpi(k)) \tag{65}$$

4. Element sizing, time integration and number of elements

The coating thickness was divided into n_e elements, at all instants of time. The elements were made non-uniform with their size rising gradually from the top to the bottom. The elements near the top were chosen to be small to capture the precipitous drop in concentration there. A benefit of using non-uniform elements is reduction in computation time. A function, $r_i=((i-1)/n_e)^2L$ where i varies from 1 to n_e+1 stretched the elements from the top to the bottom of the coating. The size of element i can be obtained by $r_{i+1}-r_\text{i}$. The exponent in the stretching function can be changed to raise or lower stretching.

Matrices ***A*** and ***B*** are computed and then Eq. (42) is expanded to generate $2n$ ODEs. To these ODEs, Eqs. (5) and (12) are appended to produce a total of $2n+2$ ODEs. The set of ordinary differential equations generated by expanding Eq. (42) was integrated by a stiff solver, *ode15s*, of MATLAB. 50 elements were taken in the present study; doubling the number to 100 changed the concentrations at all locations by <1%. Code for the drying of ternary polymer solutions is given in Appendix I. MATLAB Solution consists of two programs: one is function file and other one is main file. In main file, number of the elements, coating thickness, coating temperature and concentration or weight fractions of any two species are to be defined. In function file, all free volume parameters, mass transfer coefficients, heat transfer coefficients, size of coating, physical and chemical properties of each species and sample holder are to be defined. It generates matrices ***A***, ***B***, and ***C*** which were then integrated using MATLAB. A typical run on a 2.66 GHz computer with a memory of 506 MB took about 20 s.

Code was tested with the published results of Alsoy and Duda (1999). All the free volume parameters are given in Table 2 and experimental conditions and other parameters are given in Table 3.

Fig. 3 shows the comparison of residual solvent and temperature with time for polystyrene–toluene–tetrahydrofuran system. There is very good agreement between the simulation results of this work with earlier data (Alsoy & Duda, 1999). Hence, current formulation and solution scheme can be used for further analysis of drying process.

Figs. 4–6 show the evolution of concentration profiles during the course of drying of coating. Both the solvents evaporated from the coating into the air. Initially concentration of less volatile solvents (toluene) goes up due to higher rate of evaporation of high

Table 3
Experimental parameters for polystyrene–toluene–tetrahydrofuran system (Alsoy & Duda, 1999).

Initial conditions	Temperature	303 K
	Coating thickness	0.00577 cm
	Initial composition of solvent 1	0.321 cm^3/g
	Initial composition of solvent 2	0.321 cm^3/g
Substrate parameters	Heat capacity	1.25 J/g K
	Density	1.37 cm^3/g
	Base thickness	0.003556 cm
Coating parameters	Heat capacity	2.12 J/g K
	Density of polymer	1.083 cm^3/g
	Heat of evaporation of solvent 1	360 J/g
	Heat of evaporation of solvent 2	435 J/g
Operating conditions	Base side heat transfer coefficient, h^g	0.9228×10^{-3} W/cm^2 K
	Coat side heat transfer coefficient, h^G	10.944×10^{-4} W/cm^2 K
	Bottom air supply temperature, T^g	333 K
	Top air supply temperature, T^G	333 K
	Mass transfer coefficient of solvent 1	1.85×10^{-9} s/cm
	Mass transfer coefficient of solvent 2	1.71×10^{-9} s/cm
	Mole fraction of the solvent 1 in the air	0
	Mole fraction of the solvent 2 in the air	0

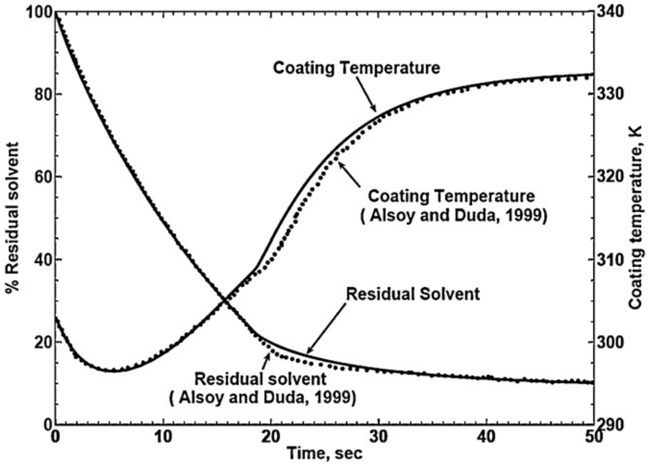


Fig. 3. Comparison of drying model predictions of this paper with published data for polystyrene–toluene–tetrahydrofuran system. Initial coating thickness was 577 μm, concentrations of toluene and tetrahydrofuran were both 0.321 g cm^{-3}. Initial coating temperature was 303 K, air temperature was 333 K and top and bottom side heat transfer coefficients were 10.944 W m^{-2} °C^{-1} and 9.228 W m^{-2} °C^{-1}, respectively.

volatile solvent (tetrahydrofuran) as shown in Fig. 5 after 5 s. After 5 s, toluene concentration starts falling and steep concentration profile is developed near the surface after 20 s of drying. Selection of unequal element size can only produce such kind of results in lesser computational time. Therefore, near the interface, element size was smaller than that of near the substrate where profile is nearly flat.

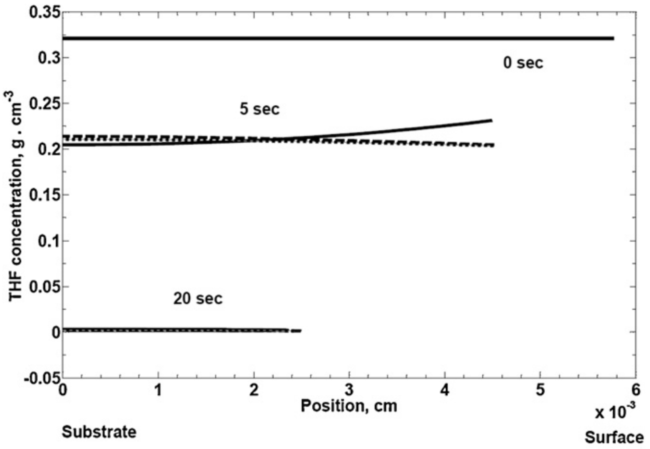


Fig. 4. Evolution of tetrahydrofuran concentration profiles in poly(styrene)–toluene–tetrahydrofuran (THF) system.

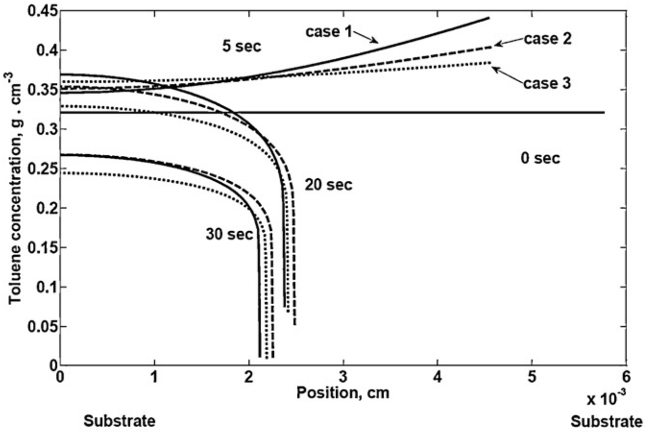


Fig. 5. Evolution of toluene profiles in poly(styrene)–toluene–tetrahydrofuran coating system.

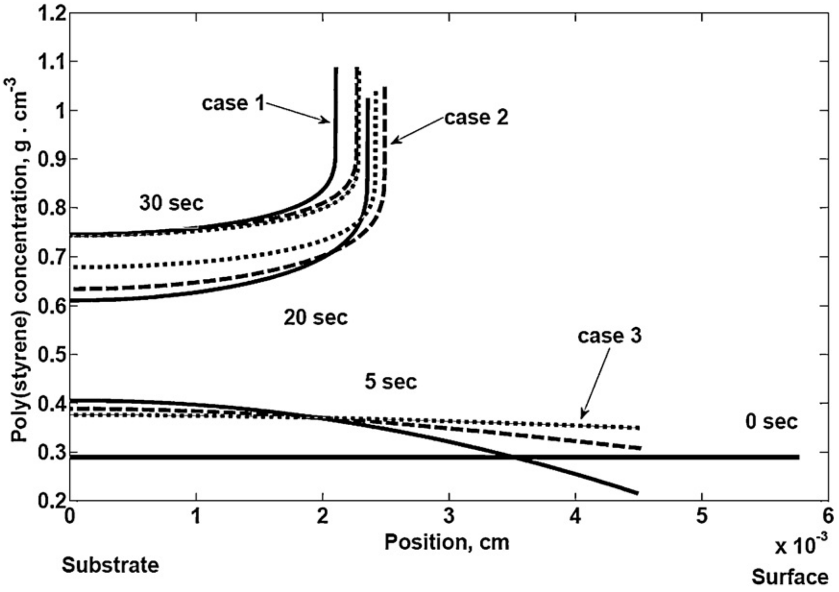


Fig. 6. Evolution of poly(styrene) profiles in poly(styrene)–toluene–tetrahydrofuran coating system.

Fig. 6 shows the profile of poly(styrene) during the course of drying. Initially concentration of poly (styrene) drop near the coating-air interface because of low external mass transfer rate. As drying proceeds, drying mechanism changes from externally controlled to internal diffusion controlled which is very slow process in case of polymeric coatings. Therefore, longer time is needed to remove the same amount of solvent at later stages.

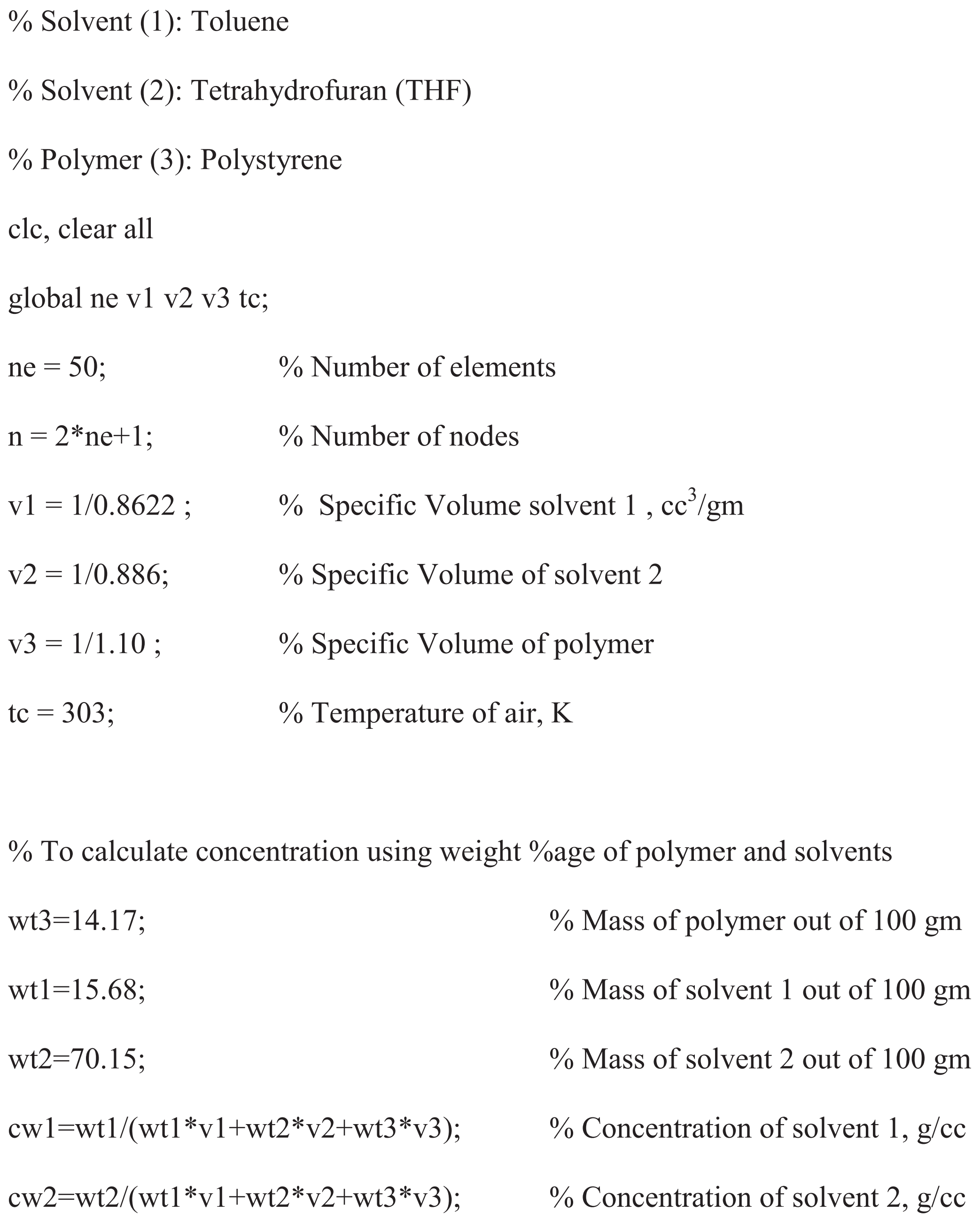
5. Conclusions

Finite element formulation using Galerkin's methods of weighted residuals has been discussed extensively for multicomponent polymer–solvent–solvent systems. Three point Gauss quadrature polynomials have been used to evaluate coefficients of matrices ***A*** and ***B***. Finite element solution has been validated with earlier published data for poly(styrene)–toluene–tetrahydrofuran system.

Acknowledgement

Author is very much thankful to Prof. Madhu Vinjamur, Department Chemical Engineering, Indian Institute of Technology, Bombay, India, for his guidance and thoughtful discussions.

Appendix I. Code for drying of polymer–solvent–solvent coating



```
% Solvent (1): Toluene

% Solvent (2): Tetrahydrofuran (THF)

% Polymer (3): Polystyrene

clc, clear all

global ne v1 v2 v3 tc;

ne = 50;            % Number of elements

n = 2*ne+1;         % Number of nodes

v1 = 1/0.8622 ;     %  Specific Volume solvent 1 , cc^3/gm

v2 = 1/0.886;       % Specific Volume of solvent 2

v3 = 1/1.10 ;       % Specific Volume of polymer

tc = 303;           % Temperature of air, K


% To calculate concentration using weight %age of polymer and solvents

wt3=14.17;                            % Mass of polymer out of 100 gm

wt1=15.68;                            % Mass of solvent 1 out of 100 gm

wt2=70.15;                            % Mass of solvent 2 out of 100 gm

cw1=wt1/(wt1*v1+wt2*v2+wt3*v3);       % Concentration of solvent 1, g/cc

cw2=wt2/(wt1*v1+wt2*v2+wt3*v3);       % Concentration of solvent 2, g/cc
```

```
cw3=wt3/(wt1*v1+wt2*v2+wt3*v3);          % Concentration of polymer, g/cc

%%%%%%%%%%  For this case use following values of concentrations

cw1=0.321;

cw2=0.321;

% Assigning Initial Concentration at All Nodes for Both the Solvents

for i=1:n

   u(i,1)=cw1;                     % Initial concentration of solvent 1 at all nodes

   u(i+n,1)=cw2;                   % Initial concentration of solvent 2 at all nodes

end

u(2*n+1,1)=0.00577;                         %  Initial coating thickness, cm

u(2*n+2,1)=tc;                     % Initial coating temperature, K


[t,u]=ode15s(@code_ps_tol_thf,[0:1:2397],u);


% To plot concentration of solvent 1 at all nodded for given drying time

figure(1)

plot(t,u(:,1:n),'b')

hold on

% To plot concentration of solvent 2 at all nodded for given drying time

figure(2)

plot(t,u(:,n+1:2*n),'b')

hold on

% To plot change in coating thickness for given drying time

figure(3)

plot(t,u(:,2*n+1),'r')

hold on
```

```
% To plot change in coating temperature for given drying time

figure(4)

plot(t,u(:,2*n+2),'r')

hold on

for i =1:length(t)

% Calculation for residual solvent starts

   c1t=0;          % set initial amount of solvent 2

   c2t=0;          % set initial amount of solvent 2

% To calculate element size

   for k =1:ne;

                    kk1(i,k)=(((k+1-1)/ne)^2 - ((k-1)/ne)^2)*u(i,2*n+1);

                   end

       x =[sqrt(15)/5,0,-sqrt(15)/5];      % points of Gauss quadrature

   w=[5/9,8/9,5/9];                        % values of weighting function


          r=0.0;

         for l=1:2:n-2

                    m =l;

                    r =r+1;

                              for j=1:3

                                       % Phis

                                        fi(m,j)=(x(j)*x(j) - x(j))/2;

                                       fi(m+1,j)=(-x(j)*x(j)+1);

                                        fi(m+2,j)=(x(j)*x(j) + x(j))/2;

                                        end

                   for k =1:3
```

```
c1t=c1t+(kk1(i,r)/2)*(u(i,m)*fi(m,k)*w(k)+u(i,m+1)*fi(m+1,k)*w(k)
    +u(i,m+2)*fi(m+2,k)*w(k));      % Amount of solvent 1 at a given time, gm
  c2t=c2t+(kk1(i,r)/2)*(u(i,m+n)*fi(m,k)*w(k)+u(i,m+1+n)*fi(m+1,k)*w(k)
    +u(i,m+2+n)*fi(m+2,k)*w(k));    % Amount of solvent 2 at a given time, gm
            end
  end
  ct(i)=c1t+c2t;                % Total amount of solvent in the coating , gm
  cwt(i) = ct(i)*100/ct(1);     % Percentage solvent remaining in the coating
  cav1(i)=c1t/u(i,2*n+1);       % Average solvent 1 within the coating
  cav2(i)=c2t/u(i,2*n+1);       % Average solvent 2 within the coating

% Calculation for residual solvent ends
% To calculate polymer concentration
  for j=1:n
    c3(i,j)=(1-u(i,j)*v1-u(i,j+n)*v2)/v3;
    c1(i,j)=u(i,j);
    c2(i,j)=u(i,j+n);
  end
% To calculate node positions if base is at zero
  for j=1:2*ne+1
    pp(i,j)=p(i,2*ne+1)-p(i,j);
  end
end

% To plot residual solvents with time
figure(5)
```

```
plot(t,cwt,'m')

hold on


% To plot residual solvent and coating temperature on primary and secondary y-axis

 figure(7)

 [AX,cwt,u(:,2*n+2)]=plotyy(t,cwt,t,u(:,2*n+2));

 set(cwt,'color','b')

 set(u(:,2*n+2),'color','b')

 set(cwt,'linestyle','-')

 set(u(:,2*n+2),'linestyle',':')

 set(cwt,'linewidth',2.5)

 set(u(:,2*n+2),'linewidth',2.5)

 hold on


% To plot average residual solvent with time

figure(8)                          % Average concentration of solvent 1 with time

  plot(t,cav1,'k')

  hold on

figure(9)                          % Average concentration of solvent 2 with time

  plot(t,cav2,'k')

  hold on

% To calculate concentration at nodes


for j = 1:50:length(t)                    % Time interval between two profiles is 50 s

   figure(6)                              % Concentration profiles of polymer

    plot(pp(j,1:n),c3(j,1:n),'k')
```

```
    hold on
    figure(7)                              % Concentration profiles of solvent 1
    plot(pp(j,1:n),u(j,1:n),'k')
    hold on
    figure(8)                              % Concentration profiles of solvent 2
    plot(pp(j,1:n),u(j,n+1:2*n),'k')
    hold on
end
% Start of calculations of model concentration at given experimental times and positions
count = 1;
texparray= time;   %%   define all times here
positionarray=position;      % define all position from top on which concentration need to be
calculated.
c1exp=toluene;   % experimental values of toluene concentration
c2exp=thf;   % experimental values of THF
c3exp=ps;   % esperimental values of poly(styrene)
for itx = 1:length(texparray)
   texp = texparray(itx);
pexp=positionarray(itx)*100/1000000;
   kkk(texp,ne+1)=0;
   for i=ne:-1:1
      kkk(texp,i)=kkk(texp,i+1)+kk1(texp,i);
      if (kkk(texp,i)==pexp)
         nn=2*i-1;
         c1model=c1(texp,nn);           % Solvent 1 concentration
         c2model=c2(texp,nn);           % Solvent 1 concentration
```

```
         c3model=c3(texp,nn);            % Polymer Concentration
      elseif (kkk(texp,i+1)<pexp && pexp<kkk(texp,i))
         nn=2*i+1;
         zeta=(pexp-pp(texp,nn))/(pp(texp,nn-2)-pp(texp,nn));
         phi1model=2*zeta^2-3*zeta+1;  % from 0 to 1
         phi2model=4*zeta-4*zeta^2;
         phi3model=2*zeta^2-zeta;
         c1model=c1(texp,nn)*phi1model+c1(texp,nn-1)*phi2model+c1(texp,nn-2)*phi3model;
         c2model=c2(texp,nn)*phi1model+c2(texp,nn-1)*phi2model+c2(texp,nn-2)*phi3model;
         c3model=c3(texp,nn)*phi1model+c3(texp,nn-1)*phi2model+c3(texp,nn-2)*phi3model;
      end
   end
   texparray(count) = texp;
   c1array(count,1) = c1model;
    c2array(count,1) = c2model;
   c3array(count,1) = c3model;
   count = count +1;
end

% End of calculations of model concentration at given experimental times and positions
% To plot experimental and model predicted concentration in same graph
figure(9)
plot(texparray,c1array,'kv')    % Plot time and model concentration of solvent 1 at experimental
positions and time
hold on
```

```
plot(texparray,c1exp,'k^')      % Plot time and measured concentration of solvent 1 at experimental
positions and time
hold on

figure(10)
plot(texparray,c2array,'kv')    % Plot time and model concentration of solvent 2 at experimental
positions and time
hold on
plot(texparray,c2exp,'k^')      % Plot time and measured concentration of solvent 2 at experimental
positions and time
hold on
figure(11)
plot(texparray,c3array,'kv')    % Plot time and model concentration of polymer at experimental
positions and time
hold on
plot(texparray,c3exp,'k^')      % Plot time and measured concentration of polymer at experimental
positions and time
hold on

% m-file called by the main program is given below
function du = code_ps_tol_thf(t,u)
global ne v1 v2 v3 tc;
global k13 k23 ze13 ze23;

%Calculation of elements sizes starts
for j = 1:ne+1;
```

```
   p(j) =[((j-1)/ne)^2 *u(2*n+1)];                %         Position of element
end
for i =1:ne;
   kk(i)=p(i+1)-p(i);                              %        Size of element
end
%Calculation of element size ends


%Interaction Parameter Input Data
del2=19.129 ;            % Solubility parameter solvent 2 , [J/cc]^1/2
del1=18.15 ;             % Solubility Parameter of solvent 1
del3=19.81 ;             % Solubility Parameter of polymer
v22=81.942;              % Molar Volume of  solvent 2 ,  cc/mol
v11=106.556;             % Molar Volume of  solvent 1 ,  cc/mol
volair=20.1;             % molar volume of air, cc/mol
mol3=230000;             % Molecular weight of polymer
mol2=72.10;              % Molecular weight of solvent 2
mol1=106.167;            % Molecular weight of  solvent 1
molair=28.97;            % molecular weight of air, g/mol
roh1=1/v1;               % Density of  solvent 1 , gm/cc
roh2=1/v2;               % Density of solvent 2 , gm/cc
roh3=1/v3;               % Density of polymer , gm/cc
v33=mol3/roh3;           % Molar Volume of polymer ,  cc/mol
R11=8.3145;              % Universal Gas Constant,  J/gm/ mol/ K


%Flory –Huggins' interaction calculation starts
x13=0.35+v11*(del1-del3)^2/(R11*tc);        % Solvent 1 and polymer
```

```
x12=0.35+v11*( del1-del2)^2 /( R11*tc);     %  Solvent 1 as the reference

x12=0.35+v22*(del1-del2)^2/( R11*tc);       % Solvent 2 as reference

x23=0.35+v22*(del2-del3)^2/( R11*tc);       % Solvent 2 and polymer

%%  If interaction parameters values are known then define above

X

%Interaction parameter calculations ends

%Free Volume Parameters

v2stt=0.899;                     %Critical molar volume of solvent 2, cc/gm

v1stt=0.917;                     %Critical molar volume of solvent 1, cc/gm

v3stt=0.850;                     %Critical molar volume of polymer, cc/gm

tg3=94+273;                      % Glass Transition temperature of polymer , K

v3j=0.6224*tg3-86.95;            % Molar volume of polymer jumping unit, cc/mol

m3j= v3j*roh3 ;                  % Molecular weight of polymer jumping unit, gm

% m3j=163.60 ;   % known here

 ze23=v2stt*mol2/(v3stt*m3j);

                                 % Can be estimated from weight loss experiments in this work

 ze13=v1stt*mol1/(v3stt*m3j);

                                 % Can be estimated from weight loss experiments in this work

%Literature values of free-volume parameters

k12= 0.000753;          % cc/gm /K

k11= 1.45e-3;

k13= 5.82e-4;                             % Can estimated from weight loss experiments in this work

ze23=0.45;

ze13=0.58;

ze33=1;

k21= -86.32;   % K
```

```
k22= 10.45;     % K

 k23= -327;     % K                % Can be estimated from weight loss experiments in this work


d02=14.4e-4 ; % cm^2/s

                                   % Cab be estimated from weight loss experiments in this work

d01=4.82e-4; %

                                   % Can be estimated from weight loss experiments in this work

k1g =1.85e-009;  s/cm

                                          % Can be estimated from solvent 1 evaporation experiments

k2g =1.71e-009; s/cm

                                          % Can be estimated from solvent 2 evaporation experiments

  HG=10.944e-4;  W/cm^2/K                 % Coating-air side heat transfer coefficient

% Lowest possible heat transfer coefficient for natural convection on coating-gas side interface

cpp=2.12; % J/g K                  % Specific heat capacity of polymer

rohp=1.083;  % g/cc                % Density of polymer

cps=1.25; % J/g K                  % Specific heat of substrate material

rohs=1.37; % g/cc                  % Density of substrate material

hs=0.003556; %cm,                  % Thickness of substrate material

tG=  333; K                         %  Temperature of coating –air side gas

tg=333;  K                         %  Temperature of gas at substrate side


% Vapour Pressure from Yaws(1999)


pv2=13.6*9.8*power(10,(34.8700-2.7523e3/u(2*n+2)-9.5958*log10(u(2*n+2))

      +1.9889e-10*u(2*n+2)+3.5465e-6*u(2*n+2)*u(2*n+2)));

                                   % Vapor Pressure of solvent 2 (THF),  Kg/( m. s^2)
```

```
pv1 = 13.6*9.8* power(10,(34.0775 - 3.0379e3/u(2*n+2) - 9.1635*log10(u(2*n+2))+1.0289e-11
*u(2*n+2) +2.7035e-6 *u(2*n+2)*u(2*n+2))); % Kg/ m. s^2

                                     % Vapor Pressure of solvent 1 (toluene) Kg/ (m. s^2)

hv2=29.81e3/mol2; % J/g      % Enthalpy of vaporization of solvent 2(THF)

hv1=35.67e3/mol1;            % Enthalpy of vaporization of solvent 1(toluene)

hv1=360;  Alsoy and Duda(1999)

hv2=435;

% Expressions for activity ( Favre et al.,1996 )

a1=exp(log(u(1)*v1)+(1-u(1)*v1-u(n+1)*v2*v11/v22)-(1-u(1)*v1-

    u(n+1)*v2)*v11/v33 + x13*power(((1-u(1)*v1-u(n+1)*v2)),2)+

    x12*power((u(n+1)*v2),2)+u(n+1)*v2*(1-u(1)*v1-u(n+1)*v2)*(x13+x12-

     x23*v11/v22));   % Activity of solvent 1(toluene)



a2=exp(log(u(n+1)*v2)+(1-u(1)*v1*v22/v11-u(n+1)*v2)-(1-u(1)*v1-

     u(n+1)*v2)*v22/v33+x23*power(((1-u(1)*v1-u(n+1)*v2)),2)+

    x12*(v22/v11)*power((u(1)*v1),2)+u(1)*v1*(1-u(1)*v1-

    u(n+1)*v2)*(x12*v22/v11+x23-x13*v22/v11)); % Activity of solvent 2(THF)


p1i=pv1*a1; % Kg/(s^2.m)     % Interfacial vapor pressure of solvent 1(toluene)

p2i=pv2*a2;                  % Interfacial vapor pressure of solvent 2(THF)



% Generation of A & B matrices starts


x =[sqrt(15)/5,0,-sqrt(15)/5];          % Gauss points
```

```
w=[5/9,8/9,5/9];                        % Weighting

r=0.0;

for i =1:2*n

   for j = 1:2*n

      f1(i,j)=0.0;               % To set all inputs of matrix A zeros

      f2(i,j)=0.0;               % To set all inputs of matrix B zeros

   end

end

for l=1:2:n-2

   m =l;

   r =r+1;

   for k=1:3

      % First Derivatives  of basis function

      df(m,k)=(x(k)*2-1)/2;

      df(m+1,k)=(-x(k)*2);

      df(m+2,k)=(x(k)*2+1)/2;

      % Phi's , basis fucntion

      fi(m,k)=(x(k)*x(k) - x(k))/2;

      fi(m+1,k)=(-x(k)*x(k)+1);

      fi(m+2,k)=(x(k)*x(k) + x(k))/2;

   end

   for k = 1:3

      c1(r,k)=u(m)*fi(m,k) +u(m+1)*fi(m+1,k)+u(m+2)*fi(m+2,k);

% Concentration of solvent 1(toluene) at nodes

      c2(r,k)=u(m+n)*fi(m,k)+u(m+1+n)*fi(m+1,k)+u(m+2+n)*fi(m+2,k);

                                 % Concentration of solvent 2(THF) at nodes
```

```
    phi1(r,k)=c1(r,k)*v1;      % Volume fraction of solvent 1(toluene) at nodes

    phi2(r,k)=c2(r,k)*v2;      % Volume fraction of solvent 2(THF) at nodes

    phi3(r,k)=1-phi1(r,k)-phi2(r,k);  % Volume fraction of polymer(PS) at nodes

    w1(r,k)=c1(r,k)/(c1(r,k)+c2(r,k)+(1-c1(r,k)*v1-c2(r,k)*v2)*rohp);

                               % Weight fraction of solvent 1(toluene) at nodes

    w2(r,k)=c2(r,k)/(c1(r,k)+c2(r,k)+(1-c1(r,k)*v1-c2(r,k)*v2)*rohp);

                               % Weight fraction of solvent 2(THF) at nodes

    w3(r,k)=1-w1(r,k)-w2(r,k);        % Weight fraction of polymer(PS) at nodes

    vfh(r,k)=w1(r,k)*k11*(k21+u(2*n+2))+w2(r,k)*k12*(k22+u(2*n+2))+

          w3(r,k)*k13*(k23+u(2*n+2)); % cc^3/gm

                               % Total free-volume for jumping

d1(r,k)=d01*exp(-(w1(r,k)*v1stt+w2(r,k)*v2stt*(ze13/ze23)

      +w3(r,k)*v3stt*ze13)/vfh(r,k)); % cm^2/s

                             % Self diffusion of solvent 1 (toluene) at Gauss points

d2(r,k)=d02*exp(-(w1(r,k)*v1stt*(ze23/ze13)+w2(r,k)*v2stt +

      w3(r,k)*v3stt*ze23)/vfh(r,k)); % cm^2/s

                             % Self diffusion of solvent 2 (THF) at Gauss points

d3(r,k)=d03*exp(-(w1(r,k)*v1stt*(ze33/ze13)+w2(r,k)*v2stt*(ze33/ze23) +

      w3(r,k)*v3stt*(ze33/ze33))/vfh(r,k)); % cm^2/s

                             % Self diffusion of polymer (PS) at Gauss points

% Derivative of activity with respect to concentration

    dlna1dc1(r,k)=v1*(1/phi1(r,k)-1+v11/v33-2*x13*(1-phi1(r,k))-phi2(r,k)*(x12-

              x13-x23*v11/v22));

    dlna1dc2(r,k)=v2*(-v11/v22+v11/v33+2*x23*phi2(r,k)*v11/v22+(1-

              phi1(r,k))*(x12-x13-x23*v11/v22));
```

```
dlna2dc1(r,k)=v1*(-v22/v11+v22/v33+2*x13*phi1(r,k)*v22/v11+(1-
                    phi2(r,k))*(x12*v22/v11-x13*v22/v11-x23));
    dlna2dc2(r,k)=v2*(1/phi2(r,k)-1+v22/v33-2*x23*(1-phi2(r,k))-
                    phi1(r,k)*(x12*v22/v11-x13*v22/v11-x23));
    dlna3dc1(r,k)=v1*(-1/phi3(r,k)+(1-v33/v11)+2*x13*(v33/v11)*phi1(r,k)+
                    phi2(r,k)*(x12*v33/v11-x12*v33/v11+x23*v33/v11));
    dlna3dc2(r,k)=v2*(-1/phi3(r,k)+(1-v33/v11)+2*x23*(v33/v11)*phi2(r,k)+
                    phi1(r,k)*(x13*v33/v11-x12*v33/v11+x23*v33/v11));


%       Case 1(Alsoy and Duda, 1999)


                      d11(r,k)=d1(r,k)*c1(r,k)*dlna1dc1(r,k);
                      d12(r,k)=d1(r,k)*c1(r,k)*dlna1dc2(r,k);
                      d21(r,k)=d2(r,k)*c2(r,k)*dlna2dc1(r,k);
                      d22(r,k)=d2(r,k)*c2(r,k)*dlna2dc2(r,k);


% Case 2   (Alsoy and Duda, 1999)


d11(r,k)=d1(r,k)*c1(r,k)*dlna1dc1(r,k);      % Main diffusion coeff. of solvent 1
d12(r,k)=0;                               % Cross term diffusion coeff. of solvent 1
d21(r,k)=0;                               % Cross term diffusion coeff. of solvent 2
d22(r,k)=d2(r,k)*c2(r,k)*dlna2dc2(r,k);   % Main term diffusion coeff. of solvent 2


% Case 3   (Alsoy and Duda, 1999)
                    d11(r,k)=d1(r,k);
                    d12(r,k)=0;
                    d21(r,k)=0;
```

```
                        d22(r,k)=d2(r,k);


% Case 4   (Alsoy and Duda, 1999)

        d11(r,k)=c1(r,k)*(1-phi1(r,k))*d1(r,k)*dlna1dc1(r,k)-

                c1(r,k)*c2(r,k)*v2*d2(r,k)*dlna2dc1(r,k);

        d12(r,k)=c1(r,k)*(1-phi1(r,k))*d1(r,k)*dlna1dc2(r,k)-

                c1(r,k)*c2(r,k)*v2*d2(r,k)*dlna2dc2(r,k);

                d21(r,k)=c2(r,k)*(1-phi2(r,k))*d2(r,k)*dlna2dc1(r,k)-

                c1(r,k)*v1*c2(r,k)*d1(r,k)*dlna1dc1(r,k);

                d22(r,k)=c2(r,k)*(1-phi2(r,k))*d2(r,k)*dlna2dc2(r,k)-

                c1(r,k)*v1*c2(r,k)*d1(r,k)*dlna1dc2(r,k);

%         Zielinski and Hanley (1999)


      d11(r,k)=d1(r,k)*c1(r,k)*(1-phi1(r,k)+c1(r,k)*v3)*dlna1dc1(r,k)+

                d2(r,k)*c1(r,k)*c2(r,k)*(v3-v2)*dlna2dc1(r,k);

       d12(r,k)=d1(r,k)*c1(r,k)*(1-phi1(r,k)+c1(r,k)*v3)*dlna1dc2(r,k)+

                d2(r,k)*c1(r,k)*c2(r,k)*(v3-v2)*dlna2dc2(r,k);

      d21(r,k)=d2(r,k)*c2(r,k)*(1-phi2(r,k)+c2(r,k)*v3)*dlna2dc1(r,k)+...

                d1(r,k)*c1(r,k)*c2(r,k)*(v3-v1)*dlna1dc1(r,k);

       d22(r,k)=d2(r,k)*c2(r,k)*(1-phi2(r,k)+c2(r,k)*v3)*dlna2dc2(r,k)+

                d1(r,k)*c1(r,k)*(v3-v1)*c2(r,k)*dlna1dc2(r,k);


% Generalized Model ( Price and Romdhane , 2003)


d11(r,k)=c1(r,k)*phi2(r,k)*(d1(r,k)*dlna1dc1(r,k)-d2(r,k)*dlna2dc1(r,k))+

            c1(r,k)*phi3(r,k)*(d1(r,k)*dlna1dc1(r,k)-d3(r,k)*dlna3dc1(r,k));
```

```
d12(r,k)=c1(r,k)*phi2(r,k)*(d1(r,k)*dlna1dc2(r,k)-d2(r,k)*dlna2dc2(r,k))+

            c1(r,k)*phi3(r,k)*(d1(r,k)*dlna1dc2(r,k)-d3(r,k)*dlna3dc2(r,k));

d21(r,k)=c2(r,k)*phi1(r,k)*(d2(r,k)*dlna2dc1(r,k)-d1(r,k)*dlna1dc1(r,k))+

            c2(r,k)*phi3(r,k)*(d2(r,k)*dlna2dc1(r,k)-d3(r,k)*dlna3dc1(r,k));

d22(r,k)=c2(r,k)*phi1(r,k)*(d2(r,k)*dlna2dc2(r,k)-d1(r,k)*dlna1dc2(r,k))+

            c2(r,k)*phi3(r,k)*(d2(r,k)*dlna2dc2(r,k)-d3(r,k)*dlna3dc2(r,k));

   end

   for i =m:m+2

      for j =m:m+2

         for k =1:3

f1(i,j)=f1(i,j)+ (kk(r)/2) * fi(i,k)*fi(j,k)* w(k);

                                    % To calculate left hand side of matrix A

            f1(i+n,j+n)=f1(i+n,j+n)+ (kk(r)/2) * fi(i,k)*fi(j,k)* w(k);

                                    % To calculate right hand side of matrix A

            f2(i,j)=f2(i,j)+  (2/kk(r)) * d11(r,k) * df(i,k)* df(j,k)*w(k);

                                    % To calculate upper left part of matrix B

            f2(i,j+n) = f2(i,j+n)+ (2/kk(r)) * d12(r,k)*df(i,k)*df(j,k)*w(k);

                                    % To calculate upper right part of matrix B

            f2(i+n,j) = f2(i+n,j)+ (2/kk(r)) * d21(r,k)*df(i,k)*df(j,k)*w(k);

                                    % To calculate lower left part of matrix B

            f2(i+n,j+n) = f2(i+n,j+n) + (2/kk(r)) * d22(r,k)*df(i,k)*df(j,k)*w(k);

                                    % To calculate lower right part of matrix B

         end

      end

   end

end
```

```
%   Boundary conditions
for i =1:2*n
   c(1)=-((1-u(1)*v1)*k1g*p1i -u(1)*v2*k2g*p2i)*10;
   c(n+1)=-((1-u(n+1)*v2)*k2g*p2i - u(n+1)*v1*k1g*p1i)*10;
   c(i)=0;
end
c=c';
% To keep time derivatives of concentrations on left hand side  du/dtt
a=inv(f1);       % To calculate A^-1
b = a*(-f2);     % To calculate A^-1 times (–Diffusion Matrix)
cc =a*c;         % Product of A*C
for i = 1:2*n
   u1(i)=u(i);
end
u1=u1';
du1=b*u1;
for i=1:2*n
   du(i)=du1(i)+cc(i);
end
du(2*n+1)=(-v1*p1i*k1g - v2*p2i*k2g)*10; % Change in Thickness cm/s
du(2*n+2)=-((HG*(u(2*n+2)-tG)+k1g*hv1*p1i*10+k2g*hv2*p2i*10+hg*(u(2*n+2)-
tg))/(rohp*cpp*du(2*n+1)+rohs*cps*hs)); % Change in Temperature,
du(2*n+2)=0; % There was no change in coating temperature in this work
du=du';     % All ODE's to be solved in column form
```

References

Alsoy, S., & Duda, J. L. (1999). Modeling of multicomponent drying of polymer films. *AIChE Journal*, *45*(4), 896–905.

Bristow, G. M., & Watson, W. F. (1958). Cohesive energy densities of polymers: Part 1. – Cohesive energy densities of rubbers by swelling measurements. *Transactions of the Faraday Society*, *54*, 1731–1741.

Dabral, M. (1999). *Solidification of coatings: Theory and modeling of drying, curing and microstructure growth*. Ph.D. Thesis. Univ. of Minnesota, Minneapolis, MN, USA.

Favre, E., Nguyen, Q. T., Clement, R., & Neel, J. (1996). Application of Flory-Huggins theory to ternary polymer–solvents equilibria: A case study. *European Polymer Journal*, *32*(3), 303–309.

Price, P. E., Jr., & Cairncross, R. A. (2000). Optimization of single-zone drying of polymer solution coatings using mathematical modeling. *Journal of Applied Polymer Science*, *78*(1), 149–165.

Price, P. E., Jr., & Romdhane, I. H. (2003). Multicomponent diffusion theory and its application to polymer–solvent systems. *AIChE Journal*, *49*(2), 309–322.

Strang, G., & Fix, G. J. (1973). *Analysis of the finite element method*. Englewood Cliff, Princeton Hall: Wellesley-Cambridge Press.

Vrentas, J. S., & Duda, J. L. (1977a). Diffusion in polymer–solvent systems. I. Reexamination of the free-volume theory. *Journal of Polymer Science Part B: Polymer Physics*, *15*, 403–416.

Vrentas, J. S., & Duda, J. L. (1977b). Diffusion in polymer–solvent systems. II. A predictive theory for the dependence of diffusion coefficients on temperature, concentration, and molecular weight. *Journal of Polymer Science Part B: Polymer Physics*, *15*, 417–439.

Vrentas, J. S., Duda, J. L., & Ling, H. C. (1984). Self-diffusion in polymer–solvent–solvent systems. *Journal of Polymer Science Part B: Polymer Physics*, *22*(3), 459–469.

Zielinski, J. M., & Hanley, B. F. (1999). Practical friction-based approach to modeling multicomponent diffusion. *AIChE Journal*, *45*(1), 1–12.