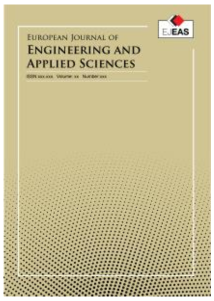
European Journal of Engineering and Applied Sciences

ISSN: 2651-3412 (Print) & 2667-8454 (Online)
Journal homepage: http://dergipark.gov.tr/EJEAS
Published by Çorlu Faculty of Engineering, Tekirdağ Namık Kemal University

European J. Eng. App. Sci. 3(1), 37-45, 2020



Research Article

A Water Level Sensor Design Using Finite Difference Solution and Its Coding in Matlab

Hakkı Canberk Karaoğlan[1]*, Reşat Mutlu[2]

Namık Kemal University, Department of Computer Engineering, Corlu, Tekirdag, Turkey[1]
Çorlu Mühendislik Fakültesi, Tekirdağ Namık Kemal Üniversitesi, Tekirdağ, Türkiye[2]

Received: 20.06.2020 *Accepted: 27.07.2020*

Abstract: Water level sensors are commonly used in industry. Some of the sensors are of capacitive nature. It is important to predict water level accurately. That's why a good sensor model is needed. The finite difference (FD) method is a simple and commonly used numerical method to solve differential equations. In literature, to the best of our knowledge, the finite difference method has not been used to design such a capacitive sensor yet. In this study, the FD method is used to solve Laplace equation in two dimensions. The variation of the sensor capacitance is obtained as a function of water level. The calculation method given here can be used as a starting point for designing such and similar sensors.

Keywords: *Water level Sensor design, Finite Difference Method, Capacitive sensor.*

Su Seviyesi Sensörünün Sonlu Farklar İle Çözümü

Özet: Su seviye sensörleri endüstride yaygın olarak kullanılır. Bazı sensörler kapasitif niteliktedir. Su seviyesini doğru tahmin etmek önemlidir. Bu yüzden iyi bir sensör modeline ihtiyaç vardır. Sonlu farklar (SF) yöntemi, diferansiyel denklemleri çözmek için yaygın olarak kullanılan basit bir sayısal yöntemdir. Literatürde, bildiğimiz kadarıyla, sonlu farklar yöntemi henüz bir kapasitif sensör tasarımında kullanılmamıştır. Bu çalışmada, SF yöntemi Laplace denklemini iki boyutta çözmek için kullanılmıştır. Sensör kapasitansındaki değişim su seviyesinin bir fonksiyonu olarak elde edilmiştir. Burada verilen hesaplama yöntemi, bu ve benzeri sensörleri tasarlamak için bir başlangıç noktası olarak kullanılabilir.

Anahtar kelimeler: *Su seviyesi Sensörü tasarımı, Sonlu Farklar Yöntemi, Kapasitif sensör.*

* Sorumlu yazar.
E-posta adresi: canberk0693@hotmail.com (H.C. Karaoğlan)

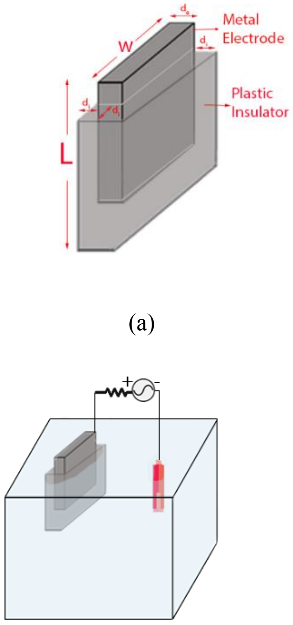
1. Introduction

In engineering, science and agriculture, it is important to measure water level and level sensors are used for this purpose [1-2]. There are different types of water level sensors. A resistive stepped transducer is analyzed and used for water level measurement in [3]. An inductive-based noncontact water level sensor is suggested in [4]. An active mode water level sensor which makes use of Villari effect has been made using magnetostrictive materials in [5]. A sound sensor is used to predict water (float) level in [6]. A fiberoptic cable is used to estimate water level and temperature [7]. Expensive methods such as Time-domain reflectometry (TDR) can also be used to measure water level [8-9]. Some of the sensors are of capacitive nature [1-2,10-11]. A capacitive sensor is a cheaper solution for water level sensing than the other methods mentioned. Accurate determination of the capacitance of rectangular parallel-plate capacitors is difficult [12]. That's why numerical methods such as Method of moments are used for 3D capacitance extraction of parallel plates [13]. Finite difference method (FDM) is also commonly used for electromagnetic problems [14-15]. A FDM for the numerical solution of three-dimensional shallow water flows is given in [16] and such methods can also be used to predict water levels in ungauged regions [17]. A capacitive sensor is hard to model due to its boundary conditions [18]. FDM can be used to analyze capacitive sensors as done in [19] for a parallel plate capacitor. That's why numerical solution of Laplace equation must be found for its solution. To the best of our knowledge, finite difference method has not been used to model a capacitive water level sensor yet. In this study, the method is used to model a capacitive sensor made of a plastic covered aluminum electrode immersed in water with water being grounded with another electrode and behaving to be an equipotential volume.

The paper is organized as follows: In the second section, sensor topology is introduced. In the third section, the finite difference (FD) solution of the system is given. In the fourth section, the simulation results are given. The paper is concluded with the last section.

2. Sensor Topology

The water level sensor topology examined in this paper is shown in Figure 1. The sensor has one electrode covered with Polyethylene (PE) nylon, which is a commonly used insulator. The other electrode touches water directly connected to the ground of the power supply feeding the capacitive sensor as shown in Figure 1.b. Since the other electrode touches the water, the water becomes an equipotential volume. Grounding a metallic box containing the water also results in the same phenomenon. The dimensions of the sensor system considered in this study is shown in Figure 1.a.



(a)

(b)

Figure 1. a) Sensor's insulated electrode and b) Sensor topology.

The sensor feeding circuit and its equivalent circuit is shown in Figure 2. The sensor can be modeled as a linear time invariant capacitor ***C*** as shown in Figure 2. However, its capacitance is dependent on the submerged depth of the sensor electrode. The resistor ***R*** shown in Figure 2 is used to measure the sensor current. The sensor capacitive reactance can be calculated as

$$X_c = \frac{1}{\omega C} \tag{1}$$

where C is the sensor capacitance and ω is the angular speed of the source voltage.

Impedance of the sensor circuit is

$$Z = \sqrt{X_c^2 + R^2} \tag{2}$$

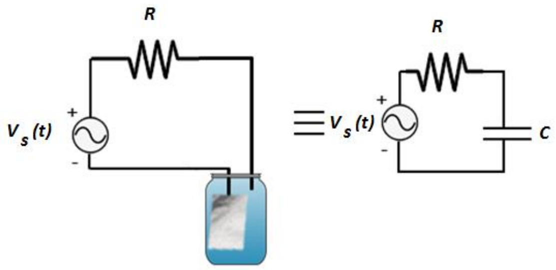


Figure 2. The sensor feeding circuit and its equivalent circuit.

If a sinusoidal source voltage of $v_S(t) = V_m \sin \omega t$ is applied to the sensor, the sensor current is given as

$$i_R = \frac{V_m}{Z}\sin\left(\omega t - \tan^{-1}\left(X_c / R\right)\right) \ . \quad (3)$$

Using the equations (1)-(3), the sensor capacitance can be measured and from the capacitance, the water level can be estimated as told in the following sections.

3. Difference Method

FD equations needed to solve the water level sensor are derived in this section. The water temperature and, therefore, its electrical permittivity is assumed to be constant for the solution. The electric field is found using the electrical potential's gradient. The charge of the rectangular prism electrode covered by PE region of a homogenous thickness is calculated from the displacement vector using Gauss Law. By using the applied voltage and the calculated charge, the sensor capacitance is calculated. Laplace equation of the electrical potential is solved in Cartesian coordinates in two dimensions by using FD. In Cartesian coordinates, when there is no volumetric charge density, the Laplacian of electrical potential in two dimensions is given as

$$\Delta V = \frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} = 0 \, . \quad (4)$$

where the x and y are dimensional components of Cartesian coordinates and $V(x,y)$ or V is the electrical potential of the point (x,y).

The sensor is placed in the Cartesian coordinate system as shown in Figure 3. Using finite difference, the second order partial derivatives of the electrical potential can be approximated as

$$\frac{\partial^2 V}{\partial x^2} \cong \frac{V(x+\Delta x, y) - 2V(x,y) + V(x-\Delta x, y)}{\Delta x^2} \quad (5)$$

and

$$\frac{\partial^2 V}{\partial y^2} \cong \frac{V(x, y+\Delta y) - 2V(x,y) + V(x, y-\Delta y)}{\Delta y^2} \quad (6)$$

where Δx and Δy are the grid intervals in x and y directions respectively.

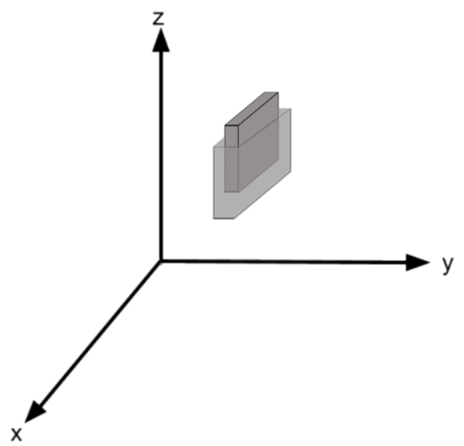


Figure 3. Placement of the sensor's insulated electrode in Cartesian coordinates.

By submitting Eq. (5) and (6) into Eq. (4):

$$\begin{aligned}\Delta V &= \frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} \\ &\cong \frac{V(x+\Delta x, y) - 2V(x,y) + V(x-\Delta x, y)}{\Delta x^2} \\ &+ \frac{V(x, y+\Delta y) - 2V(x,y) + V(x, y-\Delta y)}{\Delta y^2} = 0\end{aligned} \quad (7)$$

The electrical potential of point (x, y) can be found as

$$\begin{aligned}V(x,y) &= \frac{V(x+\Delta x, y) + V(x-\Delta x, y)}{2\Delta x^2\left(1/\Delta x^2 + 1/\Delta y^2\right)} \\ &+ \frac{V(x, y+\Delta y) + V(x, y-\Delta y)}{2\Delta y^2\left(1/\Delta x^2 + 1/\Delta y^2\right)}\end{aligned} \quad (8)$$

If the grid intervals are chosen the same, $\Delta x = \Delta y$, the electrical potential of point (x, y) becomes

$$V(x,y) = \frac{V(x+\Delta x, y) + V(x-\Delta x, y) + V(x, y+\Delta y) + V(x, y-\Delta y)}{4} \quad (9)$$

The electric field is calculated as

$$\vec{\mathbf{E}} = -grad\mathbf{V} = E_x \vec{e}_x + E_y \vec{e}_y \quad (10)$$

The electric field components in Cartesian coordinates are given as

$$E_x = -\frac{\partial V}{\partial x} \quad \text{and} \quad E_y = -\frac{\partial V}{\partial y} \quad (11)$$

The z component of the electric field is taken to be zero due to the fact that the finite difference method is applied in two dimensions to solve topology and therefore the electrical potential does not depend on z-coordinate:

$$E_z = 0 \quad (12)$$

The x and y components of the electric field can numerically be respectively approximated as

$$E_x = -\frac{\partial V}{\partial x} \cong -\frac{V(x+\Delta x, y) - V(x,y)}{\Delta x} \tag{13}$$

and

$$E_y = -\frac{\partial V}{\partial y} \cong -\frac{V(x, y+\Delta y) - V(x,y)}{\Delta y} \tag{14}$$

Considering different regions, the boundary conditions between the different regions for the electric field can be written as the follows. The normal component of the displacement vector should be the same at both sides of the boundaries between plastic and water, between water and air, and between air and water:

$$D_{n1} = D_{n2} = \varepsilon_1 E_{n1} = \varepsilon_2 E_{n2} \tag{15}$$

In addition to that, the tangential electrical field should be continuous on any boundary:

$$E_{t1} = E_{t2} \tag{16}$$

At any boundary, the normal component of the displacement vector should be continuous:

$$\varepsilon_1 \left.\frac{\partial V}{\partial n}\right|_1 = \varepsilon_2 \left.\frac{\partial V}{\partial n}\right|_2 \tag{17}$$

On the plastic and air boundary:

$$V(x,y) = \frac{\varepsilon_{plastic} V(x+\Delta x, y)/\varepsilon_{air} + V(x-\Delta x, y)}{1+\varepsilon_{plastic}/\varepsilon_{air}} \tag{18}$$

On the water and air boundary:

$$V(x,y) = \frac{\varepsilon_{water} V(x, y+\Delta y)/\varepsilon_0 + V(x, y-\Delta y)}{1+\varepsilon_{water}/\varepsilon_0} \tag{19}$$

On the plastic and water boundary:

$$V(x,y) = \frac{\varepsilon_{water} V(x+\Delta x, y)/\varepsilon_{plastic} + V(x-\Delta x, y)}{1+\varepsilon_{water}/\varepsilon_{plastic}} \tag{20}$$

Gauss-Seidel method is used for calculations. Normwise forward error is used for stability analysis and the following criterion is used to finish the calculations:

$$e_R = \frac{\left\| \underset{k}{V(i,j)} - \underset{k+1}{V(i,j)} \right\|}{\left\| \underset{k+1}{V(i,j)} \right\|} \le e_{R\max} \tag{21}$$

where the e_R is the relative error, $e_{R\max}$ is the maximum relative error, $\underset{k}{V(i,j)}$ is the potential value of point (i,j) at the ***k^th^*** iteration, and $\underset{k+1}{V(i,j)}$ is the potential value of point (i,j) at the ***k+1^th^*** iteration.

$e_{R\max}$ is taken as 0.001. However, it can be changed if desired.

After calculation of electrical potential and electric field, the sensor capacitance can be calculated. To calculate the total charge of the electrode or the charge of the sensor, Gauss's law can be used:

$$Q = \int_S \vec{D} d\vec{S} = \varepsilon \int_S \vec{E} d\vec{S} = \varepsilon \int_S E_n dS \tag{22}$$

where E_n is the normal component of the electrical field to the electrode surface.

The integral given in Eq. (22) should be solved numerically since the normal component of the electrical field is also found numerically. Considering Figure 4, the differential areas can be approximated as

$$\Delta S_y = w \Delta x \tag{23}$$

$$\Delta S_x = w \Delta y \tag{24}$$

and

$$\Delta S_z = \Delta x \Delta y \tag{25}$$

where ΔS_x is the differential area normal to x direction, ΔS_y is the differential area normal to y direction, and ΔS_z is the differential area normal to z direction.

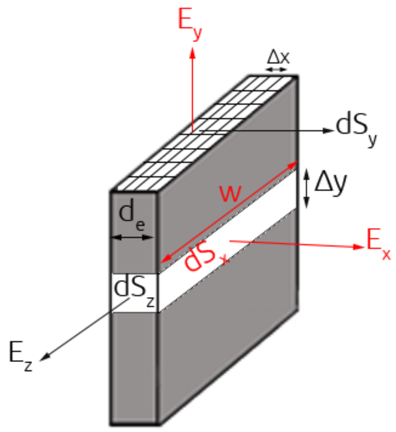


Figure 4. The differential areas used to calculate the electrode charge.

The integral given in Eq. (22) can numerically be calculated as

$$Q \cong \sum_{n=1}^{n=6} \varepsilon_n \vec{E}_n \Delta \vec{S}_n = \sum_{n=1}^{n=6} \varepsilon_n E_n \vec{n} \Delta S_n \vec{n} = \sum_{n=1}^{n=6} \varepsilon_n E_n \Delta S_n. \tag{26}$$

Since z-component of the electric field is zero and considering symmetry of the electric field, Eq. (26) can be approximated as

$$Q \cong 2\sum_{n=1}^{N} \varepsilon_x E_x \Delta S_x + 2\sum_{n=1}^{N} \varepsilon_y E_y \Delta S_y \tag{27}$$

$$Q \cong 2\sum_{n=1}^{N_y} \varepsilon_x E_x w\Delta y + 2\sum_{n=1}^{N_x} \varepsilon_y E_y w\Delta x \quad (28)$$

Where N_x is the number of points at which the x-component of the electric field are calculated, N_y is the number of points at which the y-component of the electric field are calculated, ε_x is the permittivity of the differential area ΔS_x and ε_y is the permittivity of the differential area ΔS_y .

The capacitance of the water level sensor is calculated as

$$C = \frac{Q}{V_s} \quad (29)$$

Where Vs is potential difference between the electrodes or the source voltage.

If the source voltage can be taken as 1 V, the system capacitance, in this case, becomes

$$C = Q \quad (30)$$

Dirichlet and Neumann boundary conditions are used on the outside boundaries. The number of the points needed to solve the electric field with a high accuracy are adjusted by trial and error. At the end, a system volume whose value is triple of the water volume is used for the calculations to make the leaking electric field negligible.

4. Development of the Simulation Program

In this section, the system whose electrode made of an aluminum foil wrapped with nylon to make it waterproof is dipped in water for the submerged height of h within a grounded metallic box as shown in Figure 5 is simulated. We emphasize the importance of using struct array to make definition of the system regions and parameters easier and writing of its code simpler. A struct array whose size is 101x51 is used to calculate the electrical potential within the system in this study. The electrode capacitance is calculated at 31 different water depths for each step being equal to integer times of Δy . The system topology drawn in Matlab for three different submerged heights is shown in Figure 6. Matlab allows one to draw and illustrate the system topology easily after writing a code making use of the struct array.

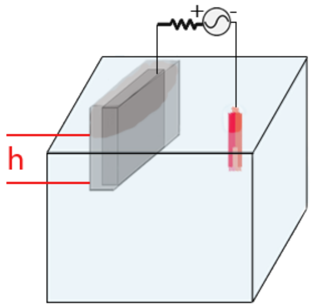
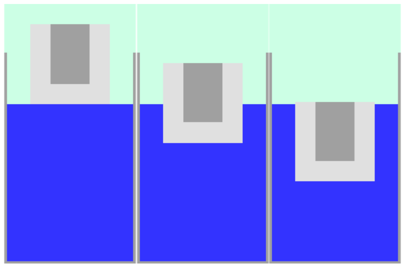


Figure 5. System topology when its electrode is at the submerged height h.



(a) (b) (c)

Figure 6. System topology drawn in Matlab at three different submerged heights:a) $h = \Delta y$,b) $h = 15\Delta y$, and c) $h = 30\Delta y$.

The development of the simulation program starts with the definition of the struct array that presents the whole system. Assigning each point to have several attributes are found to be more useful in this research and that's why Struct data type is preferred for this problem. Each point has three attributes; the surface (the medium), the potential and the permittivity, i.e., the surface identifies type of the medium and it is presented by an alphabetic character, V is potential of the point and eps (ε) is electrical permittivity of the point. The system is named as Mkap and, therefore, it can be defined as

```
Mkap(101,51) = struct('surface',' ','V',0,'eps',0);
```

After the definition of the empty struct array variable with attributes, custom written defineField function is used to define the point attributes with pre-determined coordinate values. Depth value which is an integer ranging from 0 to 30 is also used to define the location of the submerged electrode in this function:

```
Mkap = defineField (Mkap,depth);
```

The following is true:

$$h = depth * \Delta y \quad (31)$$

In this mentioned defineField function, all of the points in struct array firstly defined as Air points; then, Metal(Box), Water, Nylon and Metal(Electrode) regions are defined aftermath using loops within the code in order to avoid the writing of unnecessary or unreadable complicated code.

Since the Dirichlet boundaries has the known potentials, the potential must not be calculated at the points; a nested loop to calculate the points with the unknown potentials is required. This code also should be executed multiple times to achieve to a good accuracy using Gauss-Seidel method. The needed iteration number variable is defined as **iter** in the code. According to the explanations, the skeleton of the Finite Difference Code can be written as:

```
for it=1:iter %iteration
for i=2:100
  for j=2:50
      %Calculation Functions Should be Implemented Here
  end
end

end
```

Before examining the implementation of the calculation function, it is necessary to remember the rules that is applied in the system.

- The potentials of the submerged (aluminum) electrode which are defined as 1V at the beginning must not be calculated since they have constant values.
- Neumann Boundary Rule will be applied to the points which is on the outer edges of the air region. Potentials of the other air points, which neighbors the Neumann Boundaries, are calculated using the Finite Difference Formula and, then, the potential of a Neumann boundary point is set to be equal to that of the nearest of the air points.
- Potentials of the points within the same medium are calculated using the Finite Difference Formula.
- The potentials of the points, which are surrounded with different mediums or on the inter (interface) boundaries, are calculated using the equations (18) -(20).

For the last rule mentioned above, a function that determines the exact location of the different point near the target point by comparing nearest points and chooses the potential formula applicable to the point is found useful in this work. A custom written function called differentNeigh is written and implemented for this purpose. The target approach of the differentNeigh function is visualized for points having different boundaries as shown in Figure 7.

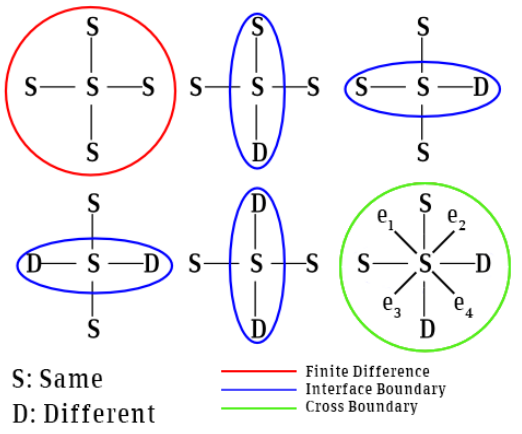


Figure 7. Visualization of the different Neigh Function Approach. differentNeigh function is shown below.

```
function [formula] = differentNeigh(x,i,j) %Written
algorithm for formula select feature
dot = x(i,j);
left = strcmp(x(i,j-1).surface,dot.surface);
right = strcmp(x(i,j+1).surface,dot.surface);
up = strcmp(x(i-1,j).surface,dot.surface);
down = strcmp(x(i+1,j).surface,dot.surface);

total = left + right + up + down;

if (total == 4)
  formula = "fdr"; %Finite Difference
elseif((total == 3 || total == 1))
  direction = [left,right,up,down];
  if (total == 1) %Will find same one (val=1)
    index = find(direction);
  else  %Total == 3, will find different one (val=0)
    index = find(~direction);
  end
  if (index==1||index==2)
    formula = "lr"; %Left Right
  else
    formula = "ud"; %Up Down
  end
elseif (total==2)
  if ((up&&down)||(left&&right)==0)
    formula ="crb"; %Cross Boundary
  else
    if (left||right)
      formula = "ud";
    else
      formula ="lr";
    end
  end
else %Unused part
  formula = "lr"; %Left Right
end
end
```

After the development of the differentNeigh algorithm, a calculation function which makes use of the differentNeigh subprogram is written and its code is given below.

```
if (strcmp(Mkap(i,j).surface,"M"))
      %Const Points
      continue;
    else
      % 40th row is the beginning of the water medium in
the metal can
      % At (39,2),(40,2),(39,50) and (40,50), interface
boundary rule
      % should be applied
      % For the rest of the edge points, finite difference
formula
      % is implemented
      boundaryCondition =
((j==2||j==50)&&(i~=39&&i~=40));
      if(boundaryCondition)
        formula ="fdr";
      else
        formula = differentNeigh(Mkap,i,j);
      end
```

```
    if (formula == "ud")
            Mkap(i,j).V =  iBoundary(Mkap(i-1,j).eps,Mkap(i+1,j).eps,Mkap(i-1,j).V,Mkap(i+1,j).V);
    elseif (formula == "lr")
        Mkap(i,j).V =  iBoundary(Mkap(i,j-1).eps,Mkap(i,j+1).eps,Mkap(i,j-1).V,Mkap(i,j+1).V);
    elseif(formula =="crb")
        Mkap(i,j).V = crossBoundary(Mkap,i,j);
    else
        Mkap(i,j).V = fDifference (Mkap(i,j-1).V,Mkap(i,j+1).V,Mkap(i-1,j).V,Mkap(i+1,j).V);
    end
    end

    if ((i>=2&&i<=19)) %Neumann Condition
      % Metal Can starts from 20th row index
      % In air medium, Neumann Boundary Condition is applied.
      % Checking row indexes range between 2nd and 20th would be
      % useful in this condition
      if(j==2)
          Mkap(i,j-1).V = Mkap(i,j).V;
      elseif(j==50)
          Mkap(i,j+1).V = Mkap(i,j).V;
      end
      if(i==2)
         Mkap(i-1,j).V = Mkap(i,j).V;
      end

    end
```

Also the predefined functions that are implemented within the calculation block are given below.

```
function [Vort] = iBoundary(e1,e2,V1,V2)
Vort = (e1*V1)/(e1+e2)+(e2*V2)/(e1+e2);
end
function [Vort] = fDifference(V1,V2,V3,V4)
Vort = (V1+V2+V3+V4)/4;
end

function [Volt] = crossBoundary(x,i,j)
e1 = x(i-1,j+1).eps; %Upper Right
e2 = x(i+1,j+1).eps;
e3 = x(i-1,j-1).eps;
e4 = x(i+1,j-1).eps;
Vup = x(i-1,j).V;
Vdown = x(i+1,j).V;
Volt = ((e1+e3)*Vup + (e4+e2)*Vdown)/(e1+e2+e3+e4);
end
```

After the implementation of the calculation functions and the calculation process of every water depth level or the calculation of the capacitance for each desired water depth level can be done. The implementation of the capacitance calculation function is written within the following block given below.

```
%Aluminium Coordinates: x:[19,33] y:[9+depth,31+depth]
```

```
function [capacitance,Etotal] =
calculateCapacitance(x,depth)
dx = (8^-1)*10^-3;dy = (8^-1)*10^-3;w = 23;
Eside = 0;Eup = 0;Edown = 0;
epsN = 4;epsA=1.000536;
eO = 8.85*10^-12;
for i=9+depth:31+depth
Eside = Eside + (abs(x(i,19)-x(i,18))/dx);
Eside = Eside + (abs(x(i,33)-x(i,34))/dx);
end
Qside = Eside*epsN*dy*w;

for i=19:33
Eup = Eup + (abs(x(9+depth,i)-x(8+depth,i))/dy);
Edown = Edown + (abs(x(31+depth,i)-x(32+depth,i))/dy);
end

Qup = epsA*Eup*dx*w;
Qdown = epsN*Edown*dx*w;
Etotal = Eup + Edown+ Eside;
capacitance = (Qside + Qup + Qdown)*eO;

end
```

For the stability analysis, the code given below is written before the end of the iteration loop.

```
%Calculation of the Error Rate Between Per Iteration
if (it == 1)
   B = convertStructToArray(Mkap,101,51);
else
   A = B;
   B = convertStructToArray(Mkap,101,51);
   errorRate = calculateER(A,B);
end

if (errorRate <= 0.001)
   disp(it);
   break;
end

function [errorRate] = calculateER (A,B)
D = abs(A-B);
errorRate = sqrt(sum(D.*D))/ sqrt(sum(B.*B));;
end

function [array] = convertStructToArray(x,ival,jval)
array = zeros(ival-1,jval-1);
for i=1:ival-1
   for j=1:jval-1
      array(i,j) = x(i,j).V;
   end
end
```

5. Simulation Results

Using the FD method, the sensor system is simulated with the programs written in Matlab and given before. The physical dimensions and electrical parameters given in Table 1 is used in all calculations. The electrical capacitance versus the submerged height of the sensor is given in Figure 7. The capacitance is not linear but it has a curved nature. Using the circuit given in Section 2, the sensor capacitance could be measured. However, due to the nonlinearity, the water sensor system would need a calibration or a look-up table system to estimate the water depth the sensor is submerged. According to simulation results, if $h < 5\Delta y$, the sensor cannot be used the estimate the submerged height since its capacitance is almost constant for $h < 5\Delta y$.

Table 1. The dimensions and the electrical parameters required for the FD analysis. The length Δy is given in the 5th section.

Sensor Parameter Symbol	Sensor Parameter Value
d, the sensor insulator thickness	8 mm
L, the electrode height	23 mm
w, the electrode width	15 mm
h_{max}, the maximum height the electrode submerged	$30\Delta y$
h, the height the electrode submerged	$0-30\Delta y$
ε_0, the permittivity of space	8.85e-12 F/m
ε_{water}, the permittivity of water	7.08e-10 F/m
$\varepsilon_{plastic}$, the permittivity of the plastic layer covering the electrode	35.4e-12 F/m (Nylon)

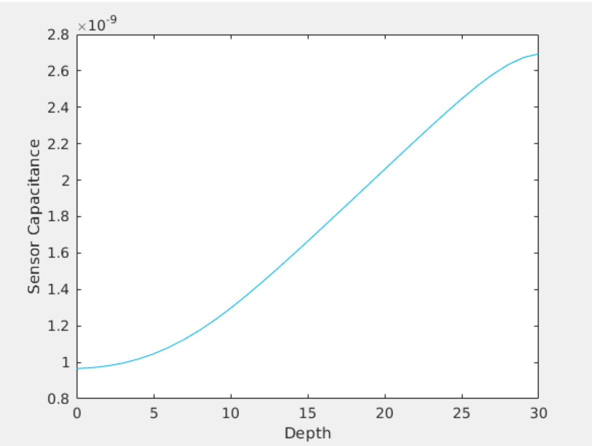


Figure 8. Sensor Capacitance versus Submerged Depth.

6. Conclusions

FD method is commonly used in examination of electromagnetic problems. It is easy to implement. The solution of Laplace equation using FD method is well-known and can be found in almost every computational electromagnetics book. A capacitive water level sensor is also an electrostatic problem. In this study, first the water level sensor topology is introduced, its is modeled with FD method considering its boundaries and different regions, its capacitance is calculated using the potentials, and, then, a Matlab code is written using struct arrays and used for the FD solution. Forward error is used to guarantee the convergence. The sensor capacitance has been found as a nonlinear function of submerged length. That's why it can only measure the depth accurately in the depths where its capacitance differs measurably, i.e. not in the depths where the capacitance is almost constant. The sensor has different regions and, in our opinion, the constant capacitance region results from the electrod being covered with PE nylon and its fringing field. The measurement system of the sensor requires some kind of calibration method either as hardware or software to suppress its nonlinearity. The FD solution given here can be used to examine sensors of similar nature. Matlab program also provides a cheap solution for FD analysis of such a sensor.

References

[1] Singh, Y., Raghuwanshi, S. K., & Kumar, S. (2019). Review on liquid-level measurement and level transmitter using conventional and optical techniques. IETE Technical Review, 36(4), 329-340.

[2] Loizou, K., & Koutroulis, E. (2016). Water level sensing: State of the art review and performance evaluation of a low-cost measurement system. Measurement, 89, 204-214.

[3] Popa, G. N., Popa, I., Diniş, C. M., & Iagăr, A. (2008, November). Resistive stepped transducer used for water level measurement. In Proceedings of the 1st WSEAS international conference on Sensors and signals (pp. 66-71). World Scientific and Engineering Academy and Society (WSEAS).

[4] Yin, W., Peyton, A. J., Zysko, G., & Denno, R. (2008). Simultaneous noncontact measurement of water level and conductivity. IEEE Transactions on Instrumentation and Measurement, 57(11), 2665-2669.

[5] Yoo, J., Jones, N. J., Flynn, K., & Jacobs, R. (2019, March). Performance of a water level sensor using magnetostrictive materials. In Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2019 (Vol. 10970, p. 109700K). International Society for Optics and Photonics.

[6] Sophia S, FLOOD ALERTING SYSTEM THROUGH WATER LEVEL METER, International Research Journal of Engineering and Technology (IRJET), Volume: 05 Issue: 03, Mar-2018, 1123-1128.

[7] Rizzolo, S., Périsse, J., Boukenter, A., Ouerdane, Y., Marin, E., Macé, J. R., ... & Girard, S. (2017). Real time monitoring of water level and temperature in storage fuel pools through optical fibre sensors. Scientific reports, 7(1), 8766.

[8] A. Thomsen, B. Hansen, K. Schelde, "Application of TDR to water level measurement," Journal of Hydrology, vol. 236, pp. 252–258, September 2000.

[9] X. Yu, X. Yu, "Laboratory Evaluation of Time-Domain Reflectioery for Bidge Scour Measurement Comparison with the Ultrasonic Method," Advances in Civil Engineering, vol. 2010, Article ID 508172, May 2010.

[10] H. Canbolat, "A Novel Level Measurement Technique Using Three Capacitive Sensors for Liquids," in IEEE Transactions on Instrumentation and Measurement, vol. 58, pp. 3762–3768, Oct. 2009.

[11] F.R.X. Li, G.C.M. Meijer, "Liquid-level measurement system based on a remote grounded capacitive sensor,"

Sensors and Actuators A: Physical, vol. 138, pp. 1–8, July 2007.

[12] Reitan, D. K. (1959). Accurate Determination of the Capacitance of Rectangular Parallel‐ Plate Capacitors. Journal of Applied Physics, 30(2), 172-176.

[13] Li, T. (2010). 3D capacitance extraction with the method of moments.

[14] Taflove, A., & Hagness, S. C. (2005). Computational electromagnetics: the finite-difference time-domain method. Artech House, Norwood.

[15] Sadiku, M. N. (2000). Numerical techniques in electromagnetics. CRC press.

[16] Casulli, V., & Cheng, R. T. (1992). Semi‐ implicit finite difference methods for three‐ dimensional shallow water flow. International Journal for numerical methods in fluids, 15(6), 629-648.

[17] Syed, Z., Choi, G., & Byeon, S. (2018). A numerical approach to predict water levels in ungauged regions—Case study of the meghna river estuary, Bangladesh. Water, 10(2), 110.

[18] Dhamodaran, M., & Dhanasekaran, R. (2013, April). Comparison of capacitance computation by different methods. In 2013 International Conference on Communication and Signal Processing (pp. 73-77). IEEE.

[19] K.P.S. Jayatilleke, Determination of the capacitance of rectangular parallel-plate capacitor using finite difference method, International Journal of Emerging Technology and Research, Volume 1, Issue 3, Mar-Apr, 2014.