Control
and Cybernetics
VOL. 3 (1974) No. 3/4

Application of numerical methods for multipole description of the electrical field of heart

by

JÓZEF JAGIELSKI
Medical School in Wrocław, Department of Pathophysiology

STANISŁAW LEWANOWICZ
University of Wrocław, Institute of Informatics

JAN MOZRZYMAS
University of Wrocław, Institute of Theoretical Physics

The paper deals with the studies on multipole components of the cardioelectric field using the multi-electrode network leads constructed following the principle of platonian polyhedra. Numerical methods were applied to develop the potential of electric field into a multipole series. Recent studies reveal that the cardioelectric field has complex structure and attempts to describe it using the dipole approximation in an oversimplification.

1. Introduction

For simplicity sake, electrocardiography assumes the electric field of the heart to have dipole nature [2].

Multipole [3] and multidipole [1] assays reveal complex structure of the electric field of the heart [13, 14].

Our studies on multipole components of the electric field of the heart using the multi-electrode network leads constructed following the principle of platonian polyhedra [10] reveal that the cardioelectric field has a complex structure and attempts to describe it using dipole approximation is an oversimplification [5, 8].

In our approach to multipole description of the electric field of the heart we are making use of the definitions accepted in the physical theory of the multipole fields [9, 15].

This theory enables description of any system of electrical charges of the heart [9].

Studies on multipole components of the electric field of the heart were carried out qualitatively and quantitatively. The simple selection rules resulting from the theory of representations of groups [4, 11] were used for qualitative studies [6, 7]. Numerical methods were applied for quantitative studies.

2. Multipole description of the electric field

Let $\sigma(x')$ denote charge density located inside of a sphere S with radius R'; $x=(x_1', x_2', x_3')$ — coordinates of a point located inside of S; r, ϑ, φ — spherical coordinates of a point $x=(x_1, x_2, x_3)$; $Y_{lm}(\vartheta, \varphi)$ — spherical functions, $l=0, 1, 2, \ldots$ $\ldots, m=-l, -(l-1), -(l-2), \ldots, -2, -1, 0, 1, 2, \ldots, l-1, l$; $q_{lm}=\int \bar{Y}_{lm}(\vartheta', \varphi')\sigma(x')$ $r'^l d^3x'$ — m-th component of 2^l-pole moment

$$Y_{l,-m}(\vartheta, \varphi)=(-1)^m \bar{Y}_{lm}(\vartheta, \varphi). \tag{1}$$

Then the potential $\Phi(x)$ at a point $x=(x_1, x_2, x_3)$, $(|x|>R')$ may be written in the form [9]

$$\Phi(x)=\sum_{l=0}^{\infty}\sum_{m=-l}^{l}\frac{4}{2l+1}q_{lm}\frac{Y_{lm}(\vartheta, \varphi)}{r^{l+1}}=\sum_{l=0}^{\infty}\sum_{m=-l}^{l}\Phi_{lm}(x). \tag{2}$$

According to the usual terminology, the terms

$$\Phi_{lm}(x)=\frac{4}{2l+1}q_{lm}\frac{Y_{lm}(\vartheta, \varphi)}{r^{l+1}}, \quad l=0, 1, 2, \ldots, \tag{3}$$

represent the 2^l-pole contribution.

In particular, for $l=0, 1, 2$ we obtain the following contributions:

(i) $l=0$, $2^0=1$ — pole (monopole) moment

$$\Phi_{00}(x)=4\pi q_{00}\frac{1}{r}Y_{00}(\vartheta, \varphi)=2\sqrt{\pi}\,q_{00}\frac{1}{r}. \tag{4}$$

(ii) $l=1$, $2^1=2$ — pole (dipole) moment

$$\Phi_{1m}(x)=\frac{4\pi}{3}q_{1m}\frac{1}{r^2}Y_{1m}(\vartheta, \varphi). \tag{5}$$

(iii) $l=2$, $2^2=4$ — pole (quadrupole) moment

$$\Phi_{2m}(x)=\frac{4\pi}{5}q_{2m}\frac{1}{r^3}Y_{2m}(\vartheta, \varphi). \tag{6}$$

It is easy to check that if we restrict ourselves to the simplest systems of electric charges treated traditionally as monopole, dipole, quadrupole etc., then the respective potential coincide with the above defined multipole potentials of the degres $l=0, 1, 2, \ldots$, etc., respectively [9].

3. Approximation of electric field potential function Program POTENTIAL (see Appendix)

Application

The program POTENTIAL is designed to approximate the $F=F(\rho, \vartheta, \varphi)$ function (ρ, ϑ, φ being the spherical coordinates of a point) with given values $F_k=$ $=F(P_k)$ upon a finite discrete set of points:

$$P_k=(r, \vartheta_k, \varphi_k), \quad k=1, 2, \ldots, n, \tag{7}$$

lying upon a sphere having the radix r and the centre in the coordinate system zero point, by means of a function

$$G_{st}(\rho, \vartheta, \varphi)=\sum_{l=s}^{t} \rho^{-l-1} w_l \sum_{m=-l}^{l} q_{lm} Y_{lm}(\vartheta, \varphi), \rho \geqslant r, \tag{8}$$

where

$$w_l=\frac{4\pi}{2l+1}$$

$$Y_{lm}(\vartheta, \varphi)=\left(\frac{2l+1}{4\pi} \frac{(l-m)!}{(l+m)!}\right)^{1/2} p_l^m(\cos \vartheta)\, e^{-im\varphi},$$

$p_l^m(x)$ is associated legendre function of the first Kind, while the coefficients q_{lm}, $l=s, s+1, \ldots, t; m=0, 1, \ldots, l$; $q_{l,-m}=(-1)^m \overline{q_{lm}}$; are determined from a condition the expression

$$R(\{q_{lm}\}) \stackrel{\mathrm{df}}{=} \sum_{k=1}^{n} [F_k - G_{st}(P_k)]^2 \tag{9}$$

to achieve the least value.

The program has been written in the language ALGOL 1204 for ODRA 1204 computer.

Method used

Let U_{lm} and V_{lm} denote real and omaginary part of q_{lm} ($q_{l0}=U_{l0}$ is real).

As (1) then the formula (8) may transformed in the following manner

$$G_{st}(\rho, \vartheta, \varphi)=\sum_{l=s}^{t} w_l \rho^{-l-1} H_l(\vartheta, \varphi),$$

where

$$H_l(\vartheta, \varphi)=U_{l0} Q_{l0}(\cos \vartheta)+2 \sum_{m=1}^{l} (U_{lm} \cos m\varphi - V_{lm} \sin m\varphi) Q_{lm}(\cos \vartheta),$$

$$Q_{lm}(x)=\left(\frac{2l+1}{4\pi} \frac{(l-m)!}{(l+m)!}\right)^{1/2} p_l^m(x).$$

The necessary condition of obtaining minimum value by the function (9) leads to the set $N \stackrel{\mathrm{df}}{=} (t+1)^2-s^2$ of linear equations with N unknowns:

$$U_{lm}, l=s, s+1, \ldots, t; m=0, 1, \ldots, l;$$

$$V_{lm}, l=s, s+1, \ldots, t; m=1, 2, \ldots, l.$$

This set is solved by elimination method with partial pivoting as described in many manuals of numerical methods.

Data

n — number of points (7)

r — radius of a sphere

ϑ_k, φ_k — spherical coordinates of the points (7)
s, t — numbers appearing in the formula (8)
D — number of the first cell of the drum area allocated for the use during the program run
S — string, comment dealing with further following data
T — number of detailed data (see note below)
F_k — values of function F in the points (7).

Data should be perforated on the tape in the following order:

$$\left.\begin{array}{l} n \quad r \\ \vartheta_1, \varphi_1, \vartheta_2, \varphi_2 \ldots \vartheta_n, \varphi_n \\ s \quad t \\ D \\ S \end{array}\right\} \tag{10}$$

$$\begin{array}{l} T\, F_1\, F_2 \ldots F_n \\ 999 \end{array} \tag{11}$$

Note. In the practice it frequently occurs that the problem of approximation is solved for an established network of nodes (7) and for many systems $\{F_k\}$. The data tape should then contain data (10) and data (11) pertaining to the first system $\{F_k\}$, second, etc. The number 999, as an accesory datum is a conventional end of data sentinel.

Results

q_{lm} — coefficients appearing in (8). The results are tabulated in $t-s+1$ lines. Moreover, a table with a heading

k pot mes pot calc,

is printed which combines the values F_k and $G_{st}(P_k)$. We gixe also the mean square error $M=R(\{q_{lm}\})^{1/2}$.

Run time

The program run time depends mainly on s, t and n. In the test runs the following times have been obtained:

n	s	t	Run time in seconds	
			for the first set F_k	for the second and next in-turn sets
30	2	2	8	2
	0	2	18	3
	0	3	61	5
	0	4	180	7
	2	4	152	6
60	2	2	14	2
	0	2	30	3
	0	3	105	5
	0	4	317	8
	2	4	276	7

Correctness check

The program was checked among others for $n=30$ and $s=t=2$. The results were obtained for which relative mean square error equalled to about 3%.

4. Conclusions

The use of numerical methods enables to develop practically the potential of cardioelectric field into a multipole series. This made it possible to determine experimentally the dominating role of miltipols components of cardioelectric field, in particular of those of rank six [16]. The multipole description gives access to new information on the heart being not revealed by dipole interpretation of electrocardiograms.

At the current status of development of the quantitative studies on the multipole components of the electric field of the heart the conclusions are of approximate character.

References

1. Barnard A. C. L., Duck I. M., Lynn M. S., The application of electromagnetic theory to electrocardiology. *Biophys. J.* **7** (1967) 443—462.
2. Burch G. E., De Pasquale N. P., A history of electrocardiography. Year Book Medical Publishers, Inc. 1964.
3. Geselowitz D. B., Multipole representation for an equivalent cardiac generator. *Proc. IRE* (1960) 75—79.
4. Hamermesh M., Group theory and its applications to physical problem. New York 1964.
5. Jagielski J., Mozrzymas J., O składnikach multipolowych pola kardioelektrycznego. *Spraw. Wroc. Tow. Nauk.* **25B** (1970) 37—39.
6. Jagielski J., Mozrzymas J., On the selection rules for the multipole generators of cardiac electric field. Wrocław 1971 Institute of Theoretical Physics, Preprint No. 230.
7. Jagielski J., Mozrzymas J., General theory of multipole generators of cardiac electric field. Proc. I European Biophysics Congr., Baden near Vienna 1971, Separatum 511—514.
8. Jagielski J., Mozrzymas J., Further study on multipole description of the cardioelectric field. Proc. Satellite Symp. of XXV Intern. Congr. of Physiological Sciences, the Electrical Field of the Heart and of the XII Intern. Col. Vectorcardiographicum. Bruxelles 1972.
9. Jackson J. D., Classi al electrodynamics. New York 1962.
10. Kowarzyk H., Kowarzykowa Z., Jagielski J., Spatial substitutes of the lead vector triangle. *Nature* **206** (1965) 1365.
11. Mozrzymas J., Application of the group theory in modern physics (in Polish). Wrocław 1967, 1970, 1971.
12. Paszkowski P., Obraz i symulacja pola kardioelektrycznego w sieci diamentoidowej. D. Sc. Thesis. Wrocław 1973.
13. Plonsey R., Theoretical consideration for a multipole probe in electrocardiographic studies. *IEEE Trans. on Bio-Medical Engng.* (1964) 105—112.
14. Plonsey R., Factors which contribute to the dipolarity of the cardiac source. Vectorcardiography 2. Amsterdam 1971.
15. Rose M. E., Multipole fields. New York 1955.
16. Utzig A., Oznaczanie składników multipolowych pola elektrycznego serca przy użyciu metod numerycznych. D. Sc. Thesis, Wrocław 1973.

APPENDIX

Program POTENTIAL for multipole description of the electrical field of the heart

```
begin
  comment Program POTENTIAL;
  integer ab1,ab2,al,al1,al2,al3,be,drumpl,drumpl1,i,i1,j,j1,k,l,l1,l2,l3,l1m,l2m,m,n,p,p1,p2,p4,t,t1,t2;
  real cabk, ctk, c0,c1,c2,clmk,fk,gj,gl,Qik,Qlm,r,s,sabk,slmk,s0,s1,s2,s3,Vk;
  read(n,r);
  begin
    array ct,f,V[1:n];
    real procedure Q(l,m,t);
      value t;
      integer l,m;
      real t;
      begin
        integer lm;
        real st,tt,tt1,c;
        real procedure P(l,m);
          integer l,m;
          if if abs(t)=1.0 then m>0 else false
            then P:=.0
            else
            begin
              integer i,lm;
              switch wl:=l0,l1,l2,l3;
              go to wl[if l≤2 then l+1 else 4];
10:           P:=1.0;
              go to endP;
11:           P:=if m=0 then t else —st;
              go to endP;
12:           P:=if m=0 then .5×(3×tt—1) else if m=1 then —3×t×st else 3×tt1;
              go to endP;
13:           P:=if m=0∨m=1 then ((l+l—1)×t×P(l—1,m)—(l+m—1)×P(l—2,m))/(l—m)
                  else —2×(m—1)×t/st×P(l,m—1)—(l—m+2)×(l+m—1)×P(l,m—2);
endP: end p;
        tt:=t×t;
        tt1:=1.0—tt;
        st:=sqrt(tt1);
        c:=1.0;
        lm:=l+m;
        for i:=1—m+1 step 1 until lm do
          c:=c×i;
        Q:=sqrt(.0795774715×(l+l+1)/c)×P(l,m)
```

```
    end Q;
  for k:=1 step 1 until n do
    begin
    read(s,ctk);
      ct[k]:=cos(ctk);
      f[k]:=s
    end k;
  read(t,p);
  t1:=t—1;
  t2:=t×t;
  p1:=p+1;
  p2:=.5×p1×(p+2)—t2;
  p4:=p1×p1—t2;
  p1:=p1—.5×(t2+t);
  format('12␣␣');
  line(4);
  if t=p
    then print('l␣=',p)
    else print('lmin␣=',t,'lmax=',p);
  begin
    integer array sub[1:p4];
    array a[1:p4,1:p4],cQ,c1Q,sQ,s1Q[1:n],g[t:p],Q0[t:p,1:n],rh,w[1:p4];
        drumpl:=drumplace:=ininteger;
        for j:=1 step 1 until p4 do
          sub[j]:=j;
        for l:=t step 1 until p do
          begin
            g[l]:=12.5663706143/(r↑(l+1)×(l+1+1));
            for m:=1 step 1 until l do
              begin
                for k:=1 step 1 until n do
                  begin
                    fk:=m×f[k];
                    Qlm:=Q(l,m,ct[k]);
                    sQ[k]:=sin(fk)×Qlm;
                    cQ[k]:=cos(fk)×Qlm
                  end k;
                todrum(n,sQ[1])
                todrum(n,cQ[1]);
              end m;
            for k:=1 step 1 until n do
              Q0[l,k]:=Q(l,0,ct[k])
          end l;
        for i1:=t step 1 until p do
          begin
            i:=i1—t1;
            for j:=t step 1 until p do
              begin
```

```
            s:=0;
            for k:=1 step 1 until n do
              s:=s+Q0[i1,k]×Q0[j,k];
            a[i,j—t1]:=g[j]×s
          end j;
        drumplace:=drumpl;
        for l:=t step 1 until p do
          begin
            l3:=.5×l×(l—1);
            l2:=p2+l3;
            l1:=p1+l3;
            gl:=2×g[l];
            for m:=1 step 1 until l do
              begin
                s:=s1:=0;
                fromdrum(n,sQ[1]);
                fromdrum(n,cQ[1]);
                for k:=1 step 1 until n do
                  begin
                    Qik:=Q0[i1,k];
                    s:=s+cQ[k]×Qik;
                    s1:=s1+sQ[k]×Qik
                  end k;
                a[i,l1+m]:=gl×s;
                a[i,l2+m]:=—gl×s1
              end m
          end l
      end i;
    drumplace:=drumpl;
    for al:=t step 1 until p do
      begin
        al3:=.5×al×(al—1);
        al1:=p1+al3;
        al2:=p2+al3;
        for be:=1 step 1 until al do
          begin
            ab1:=al1+be;
            ab2:=al2+be;
            fromdrum(n,sQ[1]);
            fromdrum(n,cQ[1]);
            drumpl1:=drumplace;
            for j1:=t step 1 until p do
              begin
                j:=j1—t1;
                s:=s1:=0;
                for k:=1 step 1 until n do
                  begin
                    Qik:=Q0[j1,k];
                    s:=s+cQ[k]×Qik;
```

```
                    s1:=s1+sQ[k]×Qik
                  end k;
                gj:=g[j1];
                a[ab1,j]:=gj×s;
                a[ab2,j]:=gj×s1
              end j;
            drumplace:=drumpl;
            for l:=t step 1 until p do
              begin
                l3:=.5×l×(l—1);
                l1:=p1+l3;
                l2:=p2+l3;
                gl:=2×g[l];
                for m:=1 step 1 until l do
                  begin
                    s:=s1:=s2:=s3:=0;
                    fromdrum(n,s1Q[1]);
                    fromdrum(n,c1Q[1]);
                    for k:=1 step 1 until n do
                      begin
                        clmk:=c1Q[k];
                        slmk:=s1Q[k];
                        cabk:=cQ[k];
                        sabk:=sQ[k];
                        s:=s+clmk×cabk;
                        s1:=s1+slmk×cabk;
                        s2:=s2+clmk×sabk;
                        s3:=s3+slmk×sabk
                      end k;
                    l1m:=l1+m;
                    l2m:=l2+m;
                    a[ab1,l1m]:=gl×s;
                    a[ab1,l2m]:=—gl×s1;
                    a[ab2,l1m]:=gl×s2;
                    a[ab2,l2m]:=—gl×s3
                  end m
              end l;
            drumplace:=drumpl1
          end be
      end al;
    for i:=1 step 1 until p4 do
      begin
        s:=0;
        for j:=i step 1 until p4 do
          begin
            s1:=abs(a[i,sub[j]]);
            if s1>s
              then
```

```
                    begin
                      s:=s1;
                      k:=j
                    end sl>gt s
                end j;
              if s=.0
                then
                begin
                  print('?macierz␣osobliwa');
                  go to ENDP
                end s:=.0;
              l1:=sub[k];
              sub[k]:=sub[i];
              sub[i]:=l1;
              s:=a[i,l1];
              i1:=i+1;
              for k:=i1 step 1 until p4 do
                begin
                  l2:=sub[k];
                  s1:=a[i,l2]:=a[i,l2]/s;
                  for j:=i1 step 1 until p4 do
                    a[j,l2]:=a[j,l2]—a[j,l1]×s1
                end k
            end i;
        end first;
        begin
        integer array title[1:100];
          instring(title[1]);
          line(10);
          outstring(title[1])
        end;
NEWDATA:
      i:=ininteger;
      if i=999
        then go to ENDP;
      read(V);
      format('␣TIME␣:␣1234');
      line(10);
      print(i);
      for i1:=t step 1 until p do
        begin
          s:=.0;
          for k:=1 step 1 until n do
        s:=s+Q0[i1,k]×V[k];
          rh[i1—t1]:=s
        end i1;
      drumplace:=drumpl;
      for a1:=t step 1 until p do
```

```
      begin
        al3:=.5×al×(al—1);
        al1:=p1+al3;
        al2:=p2+al3;
        for be:=1 step 1 until al do
          begin
            s1:=s2:=.0;
            fromdrum(n,sQ[1]);
            fromdrum(n,cQ[1]);
            for k:=1 step 1 until n do
              begin
                Vk:=V[k];
                s1:=s1+Vk×cQ[k];
                s2:=s2+Vk×sQ[k]
              end k;
            rh[al1+be]:=s1;
            rh[al2+be]:=s2
          end be
      end al;
    for i:=1 step 1 until p4 do
      begin
        l1:=sub[i];
        s:=rh[i]:=rh[i]/a[i,l1];
        for j:=i+1 step 1 until p4 do
          rh[j]:=rh[j]—a[j,l1]×s
      end i;
    for i:=p4 step —1 until 1 do
      begin
        s:=rh[i];
        for j:=i+1 step 1 until p4 do
          begin
            k:=sub[j];
            s:=s—a[i,k]×w[k]
          end j;
        w[sub[i]]:=s
      end i;
    line(4);
    print('
1       m Re q[l,m] Im q[l,m]
');
    for l:=t step 1 until p do
      begin
        format('?12␣␣12␣␣—1234.12345');
        print(l,0,w[l—t+1]);
        format('12␣␣—1234.12345␣␣—1234.12345');
        l1m:=.5×l×(l—1);
        for m:=1 step 1 until l do
          begin
            l1m:=l1m+1;
```

```
            print('? ',m,w[p1+l1m],w[p2+l1m]);
          end m;
      end l;
    format('?123␣␣—123.12␣␣—123.12');
    print('


␣␣k␣␣pot␣mes␣␣␣pot␣calc
    ');
    drumplace:=drumpl;
    for k:=1 step 1 until n do
      s1Q[k]:=.0;
        for l:=t step 1 until p do
      begin
        l1m:=.5×l×(l—1);
        for k:=1 step 1 until n do
          c1Q[k]:=.0;
        for m:=1 step 1 until l do
    begin
            l1m:=l1m+1;
            fromdrum(n,sQ[1]);
            fromdrum(n,cQ[1]);
            s1:=w[p1+l1m];
            s2:=w[p2+l1m];
            for k:=1 step 1 until n do
              c1Q[k]:=c1Q[k]+cQ[k]×s1—sQ[k]×s2
          end m;
        gl:=g[l];
        s:=w[l—t+1];
        for k:=1 step 1 until n do
          s1Q[k]:=s1Q[k]+gl×(s×Q0[l,k]+2×c1Q[k])
      end l;
    s:=.0;
    for k:=1 step 1 until n do
      begin
        s1:=s1Q[k];
        Vk:=V[k];
        print(k,Vk,s1);
        s1:=Vk—s1;
        s:=s+s1×s1
      end k;
    format('??mean␣error␣=␣␣123.12');
    print(sqrt(s/n));
    go to NEWDATA
  end;
ENDP:
  end
end
```

COMMENTS

The program has been formulated generally, and it is possible to widen its application for multipole description of the electric field. This program has been tested on the Odra-1204 computer. The input values are those of potentials measured over the sphere. The application of the above described program to the study on the cardioelectric field became possible due to potential measurements using the resistor network which reduced the measurement to the conditions of a sphere.

Practically, there were input the values measured at the points of the sphere corresponding with the vertices of dodecahedron, icosahedron, icosadodecahedron and 62-hedron being various modifications of the output of network lead system based on the principle of Platonian dual polyhedra [10]. Out of these modifications, the icosadodecahedron output point set can be obtained by means of numerical equivalent of the diamentoid network as proposed by Paszkowski [12].

Zastosowanie metod numerycznych do badania składników multipolowych pola elektrycznego serca

Omówiono badania składników multipolowych pola elektrycznego serca z użyciem wielo-elektrodowych odprowadzeń sieciowych skonstruowanych zgodnie z zasadą "figur platońskich". Do rozwinięcia potencjału pola elektrycznego w szereg multipolowy użyto metod numerycznych. Wykonane badania wykazują, że pole elektryczne serca ma strukturę złożoną i próby opisania go za pomocą aproksymacji dipolowej są nadmiernym uproszczeniem.

Применение численных методов для многополюсного описания электрического поля сердца

Статья касается исследований мультиполь овых моментов электрического поля сердца при использовании многоэлектродных сетевых отводов, построенных согласно принципу платонового многогранника.

Для разложения потенциала электрического поля в многополюсных ряд используются численные методы. Последние исследования показали, что электрическое поле сердца иммет сложную структуры и попытка описать его с помощью дипольной аппроксимации является чрезмерным упрощением.

Study support by the Medical Section of the Polish Academy of Sciences.