CHARGED PARTICLES ON A DISK: An MLAB Example

Daniel Kerner, Ph.D.
Civilized Software, Inc.
12109 Heritage Park Circle
Silver Spring MD 20906
Tel.: (301)-962-3711
email: csi@civilized.com
URL: www.civilized.com

Introduction

A classical electrostatics problem which drew some attention in the Scientific Correspondence column of *Nature* in 1985 (see *Nature*, **315** (1985) 104, 635; **316** (1985) 302; **317** (1985) 208) offers an interesting problem for the MLAB function minimization operator, `MINIMIZE`. The problem is: what will be the minimum potential energy configuration of a system of N equal point charges placed on a disk? (In what follows, the magnitude of a charge, the radius of the disk, and 4π times the capacitivity of the vacuum, ϵ, will be arbitrarily set equal to one.)

This paper will demonstrate how to use the `MINIMIZE` operator, differential equation solver, and graphics capabilities in the MLAB mathematical modeling system to find and visualize solutions to this electrostatics problem for some specific values of the number of charged particles. MLAB is a computer program whose name is an acronym for "Modeling LABoratory". It is an interactive interpreter with sophisticated instructions for performing function minimization, solving ordinary differential equations, curve fitting, and generating publication quality graphs. It was originally developed at the National Institutes of Health. It is available from Civilized Software for a variety of computer platforms including DOS- or Windows-based IBM-compatible personal computers, Macintoshes, and SUN, NeXT, IBM RS6000, DEC-Alpha and SGI Unix workstations.

Minimization with Explicit Constraints

To begin using MLAB, type MLAB at the operating system prompt of DOS or Unix systems, or double click on the MLAB icon of operating systems with a graphical user interface. MLAB then starts by typing some initialization information on the screen and an asterisk, *. The asterisk is the prompt for the user to type MLAB instructions. In what follows, MLAB instructions and responses will be typed in BOLD FACE.

The MLAB operator MINIMIZE finds a local minimum of a user defined function of several variables, optionally subject to linear constraints, given initial values of the variables. By default, a minimum or saddle point is defined as a point in parameter space where the magnitude of the components of the gradient of the objective function are less than 10^{-6}. We can use MINIMIZE to find the minimum potential energy configuration of a system of 2 equal point charges on a disk by expressing the potential energy of 2 charges in terms of the polar coordinates of each charge, defining a set of constraints that limit the polar coordinates to the unit disk, and calling the MINIMIZE operator. This simple problem will provide a basis for more complicated examples that arise later.

First, define the distance between two points on a plane given the polar coordinates of each point. If r_i and r_j are the distances of the i^{th} and j^{th} particles from the origin, and t_i and t_j are the corresponding angles measured in radians in a counter-clockwise direction from the positive x-axis, the distance between the 2 points is:

$$d(r_i, t_i, r_j, t_j) = \sqrt{(r_i \cos(t_i) - r_j \cos(t_j))^2 + (r_i \sin(t_i) - r_j \sin(t_j))^2}$$

This function can be expressed in MLAB with the FUNCTION statement (abbreviated FCT),

```
FCT D(RI,TI,RJ,TJ)=SQRT((RI*COS(TI)-RJ*COS(TJ))^2+(RI*SIN(TI)-RJ*SIN(TJ))^2)
```

For 2 charges, the potential energy of the 2 charge system is proportional to the reciprocal of this interparticle distance. We define a function for the potential energy of two particles in MLAB by typing,

```
FCT V2() = 1/D(R1,T1,R2,T2)
```

Although V2 has no arguments, it is an implicit function of the radius and angle parameters R1, R2, T1, and T2.

Next, define a set of constraints for the radii and angles. R1 and R2 must be less than or equal to 1 but greater than or equal to -1 in order for the charges to be confined to the unit disk. Since the potential energy of a configuration is the same if it is rotated by an arbitrary angle, T1 is fixed to zero. These constraints are defined with the MLAB command,

```
CONSTRAINTS Q2 = {R1<=1,R1>=-1,R2<=1,R2>=-1,T1=0}
```

To find the minimum potential energy configuration, we make initial estimates for the parameters R1,T1,R2, and T2 consistent with the constraints, and then call the minimization function.

```
R1 = .5; T1 = 0; R2 = .25; T2 = .3;
MINIMIZE(V2,R1,T1,R2,T2,Q2)
```

In this use of the MINIMIZE operator, the first argument is the name of the function to be minimized, the next 4 arguments are the names of the parameters to be varied during the minimization, and the last argument is the name of the set of linear constraints.

MLAB responds to the last command with

```
The function value is: 5.000000e-01
Argument(s): (1.000000e+00 0.000000e+00 -1.000000e+00 4.855033e-09 )
Gradient: (-2.500000e-01 -6.068791e-10 2.500000e-01 6.068791e-10 )
ecode = 0, Converged
# of function evaluations: 14
# of gradient evaluations: 28
# of Quasi-Newton iterations: 13
The following constraints are active: 5 1 4
    = 0.5
```

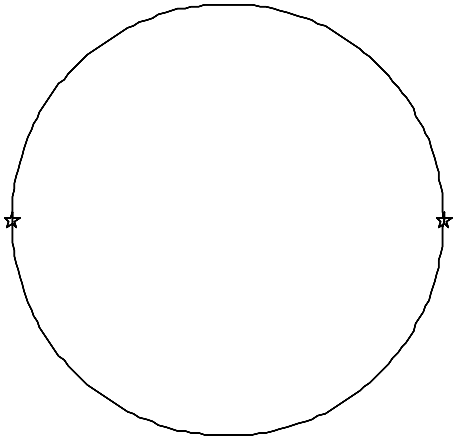
The minimum value of V2 was thus determined to be .5. In this calculation MLAB determined the symbolic gradient of the objective function and applied a Quasi-Newton method to obtain the minimum. Upon completion of the minimization, MINIMIZE reports the value of the function at the minimum, the arguments at the minimum, and the components of the gradient at the minimum. It also reports the number of times the objective function was evaluated, the number of times the gradient of the function was evaluated, and the constraints which are active at the stopping point. The value reported for ecode indicates the exit condition; 0 signifies that a minimum satisfying the exit criterion has been found.

We can visualize the results graphically by determining 200 points on a unit circle, drawing straight-lines connecting those points, and placing stars at the positions of the charges found by the minimizer. This is done as follows:

```
TL = 0:(2*PI)!200
C = (COS ON TL) &' (SIN ON TL)
```

```
NO IMAGEBOX IN W
DRAW C
WINDOW ADJUST WMATCH
DRAW (R1*COS(T1)) &' (R1*SIN(T1)) PT STAR
DRAW (R2*COS(T2)) &' (R2*SIN(T2)) PT STAR
VIEW
```

The first statement generates a row vector called TL which contains 200 equally space numbers between 0 and 2π, inclusive. The second statement generates a 2 column matrix named C which holds the x and y coordinates of the 200 points forming a circle. The third statement creates a window structure called W for drawing. The fourth statement draws the points of the circle in W. The fifth statement ensures that the aspect ratio of the window is 1 so that the curve appears circular and not elliptical. The remaining DRAW statements place stars at the positions of the charges found by the minimization process. The VIEW command then makes the following figure appear on the display.



We find, not surprisingly, that the charges occupy opposite ends of a diameter. This solution will be found for any initial estimates of the parameters R1,T1,R2, and T2 which are consistent with the constraints.

To clear the graph from the screen, type:

```
UNVIEW
DELETE W
```

Next we find the minimum potential energy configuration for 3 particles. The function to be minimized, expressed in terms of radius and angle variables R1,T1,R2,T2,R3, and T3, is defined in MLAB by the command

```
FCT V3() = 1/D(R1,T1,R2,T2)+1/D(R1,T1,R3,T3)+1/D(R2,T2,R3,T3)
```

The constraints limiting the 3 charges to the unit disk without an arbitrary rotation are defined by typing

```
CONSTRAINTS Q3 = {R1<=1,R1>=-1,R2<=1,R2>=-1,R3<=1,R3>=-1,T1=0}
```

An initial guess is made and the minimization is performed by typing:

```
R1 = 1; T1 = 0; R2 = 0; T2 = .5; R3 = 1.; T3 = PI;
MINIMIZE(V3,R1,T1,R2,T2,R3,T3,Q3)
```

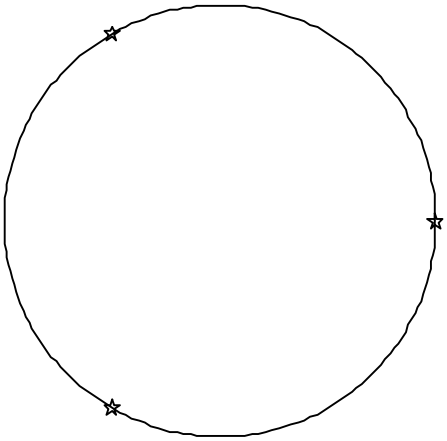
To this, MLAB responds,

```
The function value is: 1.732051e+00
Argument(s): (1.000000e+00 0.000000e+00 1.000000e+00 2.094396e+00
              1.000000e+00 4.188790e+00 )
Gradient: (-5.773502e-01 -8.001080e-08 -5.773503e-01 2.971898e-07
          -5.773503e-01 -2.171790e-07 )
ecode = 0, Converged.
# of function evaluations: 15
# of gradient evaluations: 34
# of Quasi-Newton iterations: 13
The following constraints are active: 7 1 5 3
    = 1.73205081
```

The final configuration can be visualized by typing the following commands:

```
NO IMAGEBOX IN W
DRAW C
WINDOW ADJUST WMATCH
DRAW (R1*COS(T1)) &' (R1*SIN(T1)) PT STAR
DRAW (R2*COS(T2)) &' (R2*SIN(T2)) PT STAR
DRAW (R3*COS(T3)) &' (R3*SIN(T3)) PT STAR
VIEW
```

The resulting figure reveals that the minimum potential energy configuration consists of the 3 charges at the vertices of an equilateral triangle.



The graph is cleared from the screen by typing

```
UNVIEW
DELETE W
```

The 4 particle problem is solved by typing the following statements in MLAB:

```
FCT V4() = 1/D(R1,T1,R2,T2)+1/D(R1,T1,R3,T3)+1/D(R1,T1,R4,T4)+\
           1/D(R2,T2,R3,T3)+1/D(R2,T2,R4,T4)+1/D(R3,T3,R4,T4)
CONSTRAINTS Q4 = {R1<=1,R1>=-1,R2<=1,R2>=-1,R3<=1,R3>=-1,\
           R4<=1,R4>=-1,T1=0}
R1 = 1; T1 = 0; R2 = .75; T2 = .5; R3 = .5; T3 = PI; R4 = 0; T4 = 2*PI/3;
MINIMIZE(V4,R1,T1,R2,T2,R3,T3,R4,T4,Q4)
```

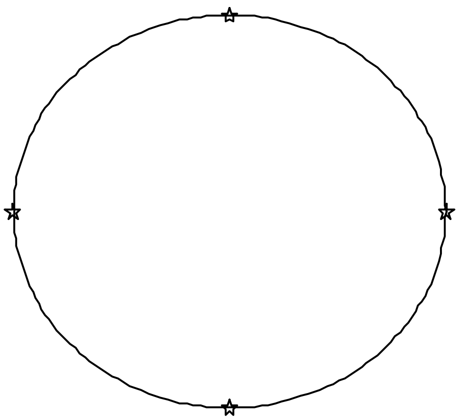
Note that the backslash character is typed to continue an MLAB command onto a second line. To this, MLAB responds

```
The function value is: 3.828427e+00
Argument(s): (1.000000e+00 0.000000e+00 1.000000e+00 3.141590e+00
              1.000000e+00 4.712386e+00 -1.000000e+00 1.099557e+01 )
Gradient: (-9.571079e-01 6.660381e-06 -9.571057e-01 2.899465e-06
           -9.571063e-01 -7.277632e-07 9.571073e-01 -8.832083e-06 )
ecode = 0, Converged.
# of function evaluations: 20
# of gradient evaluations: 56
# of Quasi-Newton iterations: 16
The following constraints are active: 9 1 3 5 8
    = 3.82842712
```

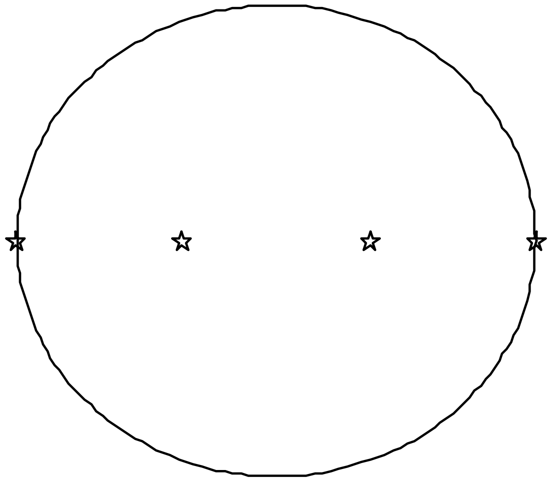
Issuing the drawing commands,

```
NO IMAGEBOX IN W
DRAW C
WINDOW ADJUST WMATCH
DRAW (R1*COS(T1)) &' (R1*SIN(T1)) PT STAR
DRAW (R2*COS(T2)) &' (R2*SIN(T2)) PT STAR
DRAW (R3*COS(T3)) &' (R3*SIN(T3)) PT STAR
DRAW (R4*COS(T4)) &' (R4*SIN(T4)) PT STAR
VIEW
```

shows the final configuration on the unit circle with the point charges occupying the vertices of a square.



The 4 particle problem demonstrates a difficulty that can arise in any greater-than-3 particle problem: the final configuration found by `MINIMIZE` may be a *local* minimum, or (rarely) a saddle point, not a *global* minimum. For example, if the initial configuration for the 4 particle problem consists of all 4 particles equally spaced along one diameter of the disk, minimization of `V4` using `MINIMIZE` leads to the following configuration with a potential energy of 6.484435:



Comparison of the potential energies for the two runs, 3.828427 to 6.484435, reveals that the configuration found in the former run is the global minimum.

The Penalty Function Method

There are N*(N-1)/2 terms in the expression for the potential energy of N equally charged particles and 2*N+1 constraints. For large N, the potential energy function and the constraint set can become cumbersome to enumerate. To simplify matters, the so-called *penalty function* method can be employed. This method consists of adding a term to the general expression for the potential energy of N point charges which causes the potential energy function to evaluate to a very large value in a continuous manner when the norm of any one of the radii is greater than 1.

The general expression for the potential energy of N point charges with 1 charge fixed at the intersection of the x-axis and the unit circle is

$$v_N(r,t) = \sum_{i=1}^{N-1}\left(\frac{1}{d(r_i,t_i,1,0)} + \sum_{j=i+1}^{N-1}\frac{1}{d(r_i,t_i,r_j,t_j)}\right)$$

where r is an array of N-1 numbers corresponding to the radii and t is an array of N-1 numbers corresponding to the angles. To this expression we add a term called a penalty term, which grows exponentially if any particle's radius should grow larger than 1. An example of the potential energy function with a penalty term which effectively confines the radii to the unit disk can be expressed in MLAB by

```
FCT VN(R,T) = SUM(I,1,N-1,1/D(R[I],T[I],1,0)+\
              SUM(J,I+1,N-1,1/D(R[I],T[I],R[J],T[J]))+\
              (IF (R[I]^2 <= 1) THEN 0 \
              ELSE 1000*(EXP(R[I]^2-1)-R[I]^2)))
```

By defining `VN` in this manner, the constraints are implicitly enforced when the function is evaluated and the potential energy function is determined for an arbitrary number of particles, N. One need only specify the value of N and initial guesses for the elements of the arrays `R` and `T` before calling the `MINIMIZE` operator. For example, the twelve particle problem may be solved by typing:

```
N = 12
T = 2*PI*(RAN ON 0^^(N-1))
R = RAN ON 0^^(N-1)
MINIMIZE(VN,R,T)
```

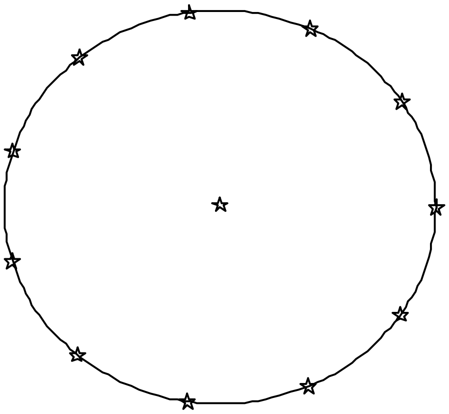
Here we have used the row-wise replication operator, which is denoted by a double caret, to create an eleven element array of zeroes. The MLAB `RAN` operator acting on the array of 11 zeroes returns 11 random numbers between 0 and 1. Thus, we are starting with a random initial configuration. MLAB's response to the `MINIMIZE` command is,

```
The function value is: 5.957297e+01
Argument(s): (1.000140e+00 1.000133e+00 1.000131e+00 1.000131e+00
              1.000141e+00 1.000153e+00 1.000137e+00 1.325854e-02
              1.000140e+00 1.000131e+00 1.000141e+00 3.421776e+00
              1.137801e+00 4.561728e+00 5.134079e+00 1.709461e+00
              5.706133e+00 2.280650e+00 1.395782e+00 2.851709e+00
              5.675664e-01 3.991057e+00 )
Gradient: (2.070053e-01 -1.246539e-01 -1.329891e-01 -1.460210e-01
           2.102167e-01 7.433105e-01 4.211969e-02 7.718492e-02
           1.650843e-01 -2.124406e-01 2.450464e-01 -2.610247e-03
          -1.696237e-02 -3.086713e-02 -1.094956e-03 8.888392e-04
          -4.157805e-02 -5.216834e-03 1.297788e-04 2.982172e-03
           -1.234044e-02 -3.144967e-02 )
ecode = 1, Exit due to slow rate of convergence.
Try a smaller tolerance value or a larger maxslow value.
# of function evaluations: 287
# of gradient evaluations: 192
# of Quasi-Newton iterations: 149
    = 59.5729652
```

In spite of the `ecode = 1` termination state, the result is the global minimum. The final configuration is visualized by typing the following MLAB commands:

```
DRAW 1&'0 PT STAR
FOR I = 1:(N-1) DO {DRAW (R[I]*COS(T[I]))&'(R[I]*SIN(T[I])) PT STAR}
WINDOW ADJUST WMATCH
VIEW
```

This results in the following figure:



From the results of 2,3, and 4 charges, you might have concluded that the minimum potential energy configuration for N charges confined to the unit disk is an N-gon inscribed on the unit circle. This conclusion is true for eleven or fewer particles, but as Berezin reported in *Nature* 315 (1985) 104, the configuration with 11 charges at the vertices of an 11-gon and one charge in the center is the minimum potential energy configuration for twelve charges. This calculation confirms Berezin's result.

The reader can use MLAB to explore minimum potential energy configurations for other values of N. We mention in passing three results reported by Queen in *Nature* 317 (1985) 208: for N=17, the lowest energy configuration consists of two charges at radius 0.31 and the other 15 charges at radius 1; for N=29, the lowest energy configuration consists of 6 charges at radius .49 and the remaining 23 at radius 1; and for N=30, the lowest energy configuration has one charge at the center, 6 at radius .55 and 23 at radius 1. These findings suggest that the charges tend to spread over the disk as N approaches infinity.

The Dynamical Problem

It is of interest to approach the problem of N charges on a unit disk from another viewpoint: what is the *dynamical motion* of N charges on a unit disk, given initial positions and velocities for each particle. In this section we will show how MLAB can be used to solve the equations of motion for five particles on the unit disk and visualize the resulting trajectory.

Let X1, Y1, X2, Y2, X3, Y3, X4, Y4, X5, and Y5 be the Cartesian coordinates, and PX1, PY1, PX2, PY2, PX3, PY3, PX4, PY4, PX5, and PY5 be the components of velocity for particles 1 through 5. We begin by defining a function for the polar radius of a particle in terms of the Cartesian coordinates of the particle by typing:

```
FCT RV(X,Y) = SQRT(X*X+Y*Y)
```

Ignoring contributions to forces on particles arising from magnetic fields (i.e. forces computed as the cross product of a particle's velocity and the magnetic field, $\vec{v} \times \vec{B}$), we assume the force on the i^{th} charge due to the j^{th} charge is given by Coulomb's law,

$$\vec{F}_{i,j} = \frac{\vec{r}_i - \vec{r}_j}{\sqrt{(\vec{r}_i - \vec{r}_j) \cdot (\vec{r}_i - \vec{r}_j)}^{\,3}}$$

where $\vec{r}_i$ is the vector from the origin to particle i and $\vec{r}_j$ is the vector from the origin to particle j. From Newton's Law, we can write an equation for the time derivative of each component of velocity for each particle:

```
FCT F(UI,VI,UJ,VJ) = (UI-UJ)/(SQRT((UI-UJ)^2+(VI-VJ)^2))^3
FCT PX1'T(T) = F(X1,Y1,X2,Y2)+F(X1,Y1,X3,Y3)+F(X1,Y1,X4,Y4)+F(X1,Y1,X5,Y5)
FCT PY1'T(T) = F(Y1,X1,Y2,X2)+F(Y1,X1,Y3,X3)+F(Y1,X1,Y4,X4)+F(Y1,X1,Y5,X5)
FCT PX2'T(T) = -F(X1,Y1,X2,Y2)+F(X2,Y2,X3,Y3)+F(X2,Y2,X4,Y4)+F(X2,Y2,X5,Y5)
FCT PY2'T(T) = -F(Y1,X1,Y2,X2)+F(Y2,X2,Y3,X3)+F(Y2,X2,Y4,X4)+F(Y2,X2,Y5,X5)
FCT PX3'T(T) = -F(X1,Y1,X3,Y3)-F(X2,Y2,X3,Y3)+F(X3,Y3,X4,Y4)+F(X3,Y3,X5,Y5)
FCT PY3'T(T) = -F(Y1,X1,Y3,X3)-F(Y2,X2,Y3,X3)+F(Y3,X3,Y4,X4)+F(Y3,X3,Y5,X5)
FCT PX4'T(T) = -F(X1,Y1,X4,Y4)-F(X2,Y2,X4,Y4)-F(X3,Y3,X4,Y4)+F(X4,Y4,X5,Y5)
FCT PY4'T(T) = -F(Y1,X1,Y4,X4)-F(Y2,X2,Y4,X4)-F(Y3,X3,Y4,X4)+F(Y4,X4,Y5,X5)
FCT PX5'T(T) = -F(X1,Y1,X5,Y5)-F(X2,Y2,X5,Y5)-F(X3,Y3,X5,Y5)-F(X4,Y4,X5,Y5)
FCT PY5'T(T) = -F(Y1,X1,Y5,X5)-F(Y2,X2,Y5,X5)-F(Y3,X3,Y5,X5)-F(Y4,X4,Y5,X5)
```

The time derivative of each particle's Cartesian coordinate is set equal to the corresponding component of the velocity so long as the polar radius of the particle is less than one. If the polar radius of the particle becomes greater than one, we constrain the particle to the unit disk by setting the radial component of the velocity equal to zero, preserving only the angular component of the velocity. The function AN defined below performs the necessary projection of the velocity vector of each particle. This is expressed in MLAB by typing:

```
FCT G(X,Y,PX,PY) = X*PX+Y*PY
FCT AN(X,Y,PX,PY,R) = PX-G(X/R,Y/R,PX,PY)*X/R
FCT K(X,Y,R,VX,VY,V) = IF (R < 1) THEN V ELSE AN(X,Y,VX,VY,R)
FCT H(X,Y,VX,VY,V) = K(X,Y,RV(X,Y),VX,VY,V)
FCT X1'T(T) = H(X1,Y1,PX1,PY1,PX1)
FCT Y1'T(T) = H(Y1,X1,PY1,PX1,PY1)
FCT X2'T(T) = H(X2,Y2,PX2,PY2,PX2)
FCT Y2'T(T) = H(Y2,X2,PY2,PX2,PY2)
FCT X3'T(T) = H(X3,Y3,PX3,PY3,PX3)
FCT Y3'T(T) = H(Y3,X3,PY3,PX3,PY3)
FCT X4'T(T) = H(X4,Y4,PX4,PY4,PX4)
FCT Y4'T(T) = H(Y4,X4,PY4,PX4,PY4)
FCT X5'T(T) = H(X5,Y5,PX5,PY5,PX5)
FCT Y5'T(T) = H(Y5,X5,PY5,PX5,PY5)
```

The initial positions and velocities are defined using the INITIAL statement which is abbreviated INIT. As a first example, we initialize four particles to the minimum potential energy configuration found for 4 particles; each particle with zero velocity. The fifth particle is placed slightly to the right of center with an initial velocity directed toward the charge lying on the negative y-axis.

```
INIT X1(0) = 1.; INIT Y1(0) = 0.; INIT PX1(0) = 0.; INIT PY1(0) = 0.;
INIT X2(0) = 0.; INIT Y2(0) = 1.; INIT PX2(0) = 0; INIT PY2(0) = 0;
INIT X3(0) = -1.; INIT Y3(0) = 0.; INIT PX3(0) = 0; INIT PY3(0) = 0;
INIT X4(0) = 0.; INIT Y4(0) = -1; INIT PX4(0) = 0.; INIT PY4(0) = 0;
INIT X5(0) = .30; INIT Y5(0) = .0; INIT PX5(0) = -.1; INIT PY5(0) = -.5;
```

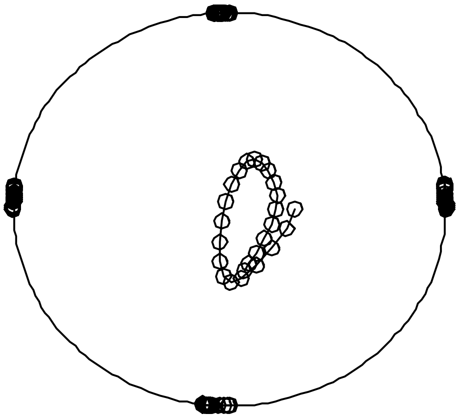
Finally we define the time range and time steps for which the motion is desired and some particulars regarding the differential equation solving method by typing:

```
JACSW = 0
T = 0:5!100
MANDSW = 1
METHOD = ADAMS
M = POINTS(X1,Y1,PX1,PY1,X2,Y2,PX2,PY2,X3,Y3,PX3,PY3,X4,Y4,PX4,PY4,\
           X5,Y5,PX5,PY5,T)
```

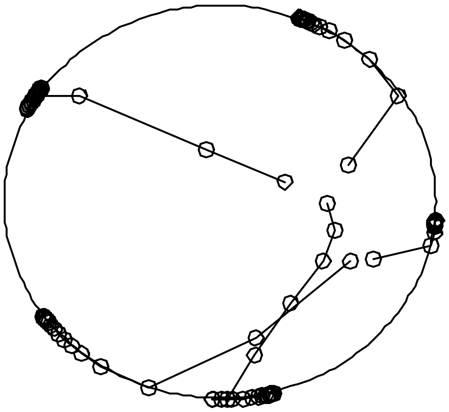
The POINTS operator integrates the equations of motion from time 0 to time 5 in 100 steps using the Adams method. It returns a 21 column by 100 row array with time in column 1 and the coordinates and components of each particle's velocity in the remaining columns. To visualize the trajectory we type:

```
DELETE M ROW 2:NROWS(M):2; DELETE M ROW 2:NROWS(M):2;
NO IMAGEBOX IN W
DRAW M COL (2,3) PT CIRCLE IN W
DRAW M COL (6,7) PT CIRCLE IN W
DRAW M COL (10,11) PT CIRCLE IN W
DRAW M COL (14,15) PT CIRCLE IN W
DRAW M COL (18,19) PT CIRCLE IN W
DRAW (COS ON 0:(2*PI)!200) &' (SIN ON 0:(2*PI)!200) IN W
WINDOW ADJUST WMATCH IN W
```

This results in the following figure:



When we choose more arbitrary initial positions and initial velocities for the charges, we obtain the following result:



Note how some charges move to the boundary, begin to travel in a direction determined by their velocity vector, and then reverse their directions under the influence of the other particles.

Such studies raise some interesting questions concerning dynamical systems. In the first example, how long will the particle in the center oscillate about the center? Will it ever escape to the boundary? Given enough time, will the particles in both examples converge to the minimum potential energy configuration for five charges? Note that this model dissipates energy due to the truncation of radial velocity components at the boundary. However the kinetic energy does not necessarily become 0; if the total angular momentum of the initial configuration is non-zero, the particles can enter a limit cycle in which they rotate forever.

Conclusion

This paper has demonstrated how MLAB can be used to determine minimum potential energy configurations and the dynamical motion of N charged particles on the unit disk.

There are related problems which might be studied using MLAB. For example, we have seen that as N increases from 11 to 12, the minimum $\frac{1}{r}$-potential energy configuration expelled a particle to the center of the disk. At what point would the minimum potential energy configuration expel a charge to the center if the inter-particle potential were of another form?

Another interesting problem one might study with MLAB is Thomson's problem: what is the minimum potential energy configuration for N charges constrained to the surface of a unit sphere?

Further information about MLAB is available from:

Civilized Software, Inc.
12109 Heritage Park Circle
Silver Spring, MD 20906 USA
Tel. (301) 962-3711
Email: `csi@civilized.com`