Information Systems (308-617B)

Notes on Constraints: Sketchpad

Copyright ©T. H. Merrett, 1991

Introduction

Much of the present literature on constraints cites Ivan Sutherland's Ph.D. thesis of almost thirty years ago [3] as a seminal work. This thesis is easy to read, but you only find out how comprehesive and valuable his system is when you fill in the missing details. Here is a discussion of the constraint system in Sketchpad.

First, an introduction. Sketchpad is a drawing system, using a lightpen, screen, and 32 special buttons on an innovative computer with 64K words (at 36 bits per word) of core memory with a 6 μsec access time, and magnetic tape secondary memory — a kind of super Apple][. The system can use the buttons to select drawing modes (straight line, circle [1]), to copy or delete, or to move data to plotter or to and from tape. More important than this, Sketchpad permits the user easily to define *constraints* on the components of the drawing, which are thereafter automatically satisfied. Sketchpad also maintains a library of both drawings and constraints, so the system grows more useful the more it is used. [2]

For instance, to draw a parallelogram, the four vertices may be placed in roughly their right positions using the light pen, then constrained to make the parallelogram. (Actually, one would draw the four lines, but it is the vertices that are constrained.) We can use two constraints to do this. The first constrains the distance from vertex 2 to vertex 3 to equal the distance from vertex 0 to vertex 1. The second constrains 2 and 3 so that a line connecting them would be parallel to a line from 0 to 1.

Both of these are quaternary constraints: they have four vertices as arguments. Sketchpad allows unary, binary, ternary, and quaternary constraints.

As another example, to draw a hexagon, draw a circle, then a six-sided polygon within it, and constrain each vertex to lie on the circle (that is, to be pairwise cocircular). Then constrain each pair of sides to be the same length (the same equality constraint used for the parallelogram).

The constraint mechanism in Sketchpad is such that it can actually be used to display the strains of a truss bridge under load, and the corresponding forces. However, Sketchpad was not able to show currents or voltages in an electrical circuit when drawn. This illustrates the generality of the constraint mechanism used, although it was only designed to aid drawing.

Constraints

Sketchpad will accept any constraint whose degree of dissatisfaction (called the *error*) can be measured by an expression returning a real number. If the error is linear in the variables to be changed, Sketchpad can satisfy the constraint (as far as possible, considering other constraints) in a single calculation. For nonlinear constraints, the satisfaction process must iterate.

An example of a linear constraint is collinearity. Given three points, $(x_0, y_0), (x_1, y_1)$ and (x_2, y_2), the error is linear in each of the variables, and can be written as the area of the triangle formed by the three points.

$$\text{error} = \begin{vmatrix} x_0 & y_0 & 1 \\ x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \end{vmatrix}$$

On the other hand, the constraint that distances between pairs of points be equal is quadratic, not linear. The error can be given as the difference of the squares of the distances.

[1] The difference equations for a line are

$$x_i = x_{i-1} + \Delta x \qquad y_i = y_{i-1} + \Delta y$$

and those for a circle are

$$x_i = x_{i-1} + (y_{i-1} - y_i)/R \qquad y_i = y_{i-1} - (x_{i-1} - x_i)/R$$

or, on a machine such as the TX-2, which can compute coordinates in parallel,

$$x_i = x_{i-2} + 2(y_{i-1} - y_i)/R \qquad y_i = y_{i-2} - 2(x_{i-1} - x_i)/R$$

[2] In a number of ways, Sketchpad is a precursor of the object-oriented approach, using complex objects and inheritance from general to specific. Sutherland considers nested instances and genericity to be the most important contributions of Sketchpad.

$$\text{error} = (x_3 - x_2)^2 + (y_3 - y_2)^2 - (x_1 - x_0)^2 - (y_1 - y_0)^2$$

The user must define error functions for each constraint introduced. There must be one error function for each degree of freedom removed by the constraint.

Linearization

Sketchpad uses the error function to produce a linear equation for each constraint. The set of equations generated for a set of constraints can be solved in standard ways, discussed below. The linear equations are derived from

$$\sum_{i=1}^{n} \frac{\partial E}{\partial x_i}(x_i - x_{i0}) = -E_0 \tag{1}$$

where $x_i, i = 1 \ldots n$ are the variables (coordinates) to be changed to meet the constraints, x_{i0} are the corresponding initial values, and E_0 is the error when the variables have their initial values.

The error function is used to estimate the partial derivatives, $\partial E/\partial x_i$, by averaging $\Delta E/\Delta x_i$ as the x_i are each independently varied from x_{i0}.

If the error function is *linear*, the linear equation produced is identical to the equation, *error* $= 0$. As an example, we find the intersection of two lines, using two collinearity constraints. The lines are defined by end points

$$(x_0, y_0) = (0, 0); (x_1, y_1) = (1, 1)$$
$$(x_2, y_2) = (1, 0); (x_3, y_3) = (0, 1)$$

and the variable point, constrained to be collinear with both lines, starts at

$$(x, y) = (0.5, 0).$$

The errors are

$$\text{error } 01 = \begin{vmatrix} 0 & 0 & 1 \\ 1 & 1 & 1 \\ x & y & 1 \end{vmatrix} = y - x \text{ and}$$

$$\text{error } 23 = \begin{vmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ x & y & 1 \end{vmatrix} = 1 - x - y.$$

Now we vary x and y about their initial values, and compute $\Delta x, \Delta y$ and ΔE

				01 collinear				23 collinear			
x	y	Δx	Δy	E	ΔE	$\frac{\Delta E}{\Delta x}$	$\frac{\Delta E}{\Delta y}$	E	ΔE	$\frac{\Delta E}{\Delta x}$	$\frac{\Delta E}{\Delta y}$
.5	0	—	—	-.5	—	—	—	.5	—	—	—
0	0	-.5	—	0	.5	-1	—	1	.5	-1	—
1	0	.5	—	-1	-.5	-1	—	0	-.5	-1	—
.5	.5	—	.5	0	.5	—	1	0	-.5	—	-1
.5	1	—	1	.5	1	—	1	-.5	-1	—	-1

We see that $(\frac{\Delta E}{\Delta x}, \frac{\Delta E}{\Delta y})$ is (-1, 1) for the 01 collinearity constraint, and (-1, -1) for the 23 collinearity constraint. Using these values for $(\frac{\partial E}{\partial x}, \frac{\partial E}{\partial y})$ in equation 1, we get the equation $y - x = 0$ for 01 collinearity and $1 - x - y = 0$ for 23 collinearity: just the original error expressions set to zero. Solving these two equations simultaneously gives $(x, y) = (0.5, 0.5)$ as the intersection point, exactly the right answer.

For *nonlinear error functions*, we must put up with approximations. But the same procedure applies, which is why I have illustrated it even in the linear case. Consider the example of making a parallelogram. We have sides

$$(x_0, y_0) = (0, 0) \qquad (x_1, y_1) = (0, 1)$$
$$(x_2, y_2) = (1, 0) \qquad (x_3, y_3) = (.5, .5)$$

and we want to constrain (x, y), initially equal to (x_3, y_3), so that line 23 is a) parallel to and b) the same length as line 01.

The parallelism constraint is linear: we just require (x, y) to be collinear with line $(x_2, y_2), (x_4, y_4)$, where (x_4, y_4) is a new point arrived at by translating (x_1, y_1) the same way that would move (x_0, y_0) to (x_2, y_2), *i.e.*, $x_4 = x_1 + x_2 - x_0$, and $y_4 = y_1 + y_2 - y_0$.

$$\text{error par} = \begin{vmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ x & y & 1 \end{vmatrix} = x - 1.$$

The equality constraint requires us to change the length of line 23 to make it equal to the length of line 01. The equality error is quadratic, the length of line 01 subtracted from the square of the distance of (x, y) from (x_2, y_2).

$$\text{error eq} = (x-1)^2 + y^2 - 1,$$

Here is the linearization.

				parallel				equal length			
x	y	Δx	Δy	E	ΔE	$\frac{\Delta E}{\Delta x}$	$\frac{\Delta E}{\Delta y}$	E	ΔE	$\frac{\Delta E}{\Delta x}$	$\frac{\Delta E}{\Delta y}$
.5	.5	—	—	-.5	—	—	—	-.5	—	—	—
1	.5	.5	—	0	.5	1	—	-.75	-.25	-.5	—
0	.5	-.5	—	-1	-.5	1	—	.25	.75	-1.5	—
.5	1	—	.5	-.5	0	—	0	.25	.75	—	1.5
.5	0	—	-.5	-.5	0	—	0	-.75	-.25	—	.5

This gives us, as we expect, the exact equation for the linear constraint, $x = 1$. The average value for $(\frac{\Delta E}{\Delta x}, \frac{\Delta E}{\Delta y})$ is (-1,1), giving, from equation 1, the equation $-1(x - .5) + 1(y - .5) = .5$, or $-x + y = 1.5$.

Solving the two equations gives $(x, y) = (1, 1.5)$, which we see is not the correct answer, $(x, y) = (1, 1)$. So we do a second iteration, starting this time at (1,1.5), the previous answer.

				parallel				equal length			
x	y	Δx	Δy	E	ΔE	$\frac{\Delta E}{\Delta x}$	$\frac{\Delta E}{\Delta y}$	E	ΔE	$\frac{\Delta E}{\Delta x}$	$\frac{\Delta E}{\Delta y}$
1	1.5	—	—	0	—	—	—	1.25	—	—	—
.5	1.5	-.5	—	-.5	-.5	1	—	1.5	.25	-.5	—
1.5	1.5	.5	—	.5	.5	1	—	1.5	.25	.5	—
1	1	—	-.5	0	0	—	0	0	-1.25	—	2.5
1	2	—	.5	0	0	—	0	3	1.75	—	3.5

The parallelism error is the same as before. The equality error becomes $3(y - 1.5) = -1.25$. Solving these together gives $(x, y) = (1, 13/12)$. This is better than (1, 1.5), but not all the way yet. We are getting closer.

Thus we see the nature of the process for nonlinear constraints, and get a feel for the rate of convergence in a simple case. Of course, we could have solved the parallel + equal length combination exactly since a linear equation and a quadratic equation in two variables gives no problem. But such a solution would not be general.

Underdetermined Equations

The two examples in the previous section have unique solutions (apart from signs on square roots in quadratic equations). Some problems will be underdetermined, as in the case of constraining a point to be collinear with a single line. In this case, Sketchpad minimizes the displacement of the point from its original position. Sutherland does not say how he does this, so I will use the method of Lagrange multipliers [1]. This method finds the minimum of a function $f(x, y, \ldots)$ with respect to $x, y, \ldots$ subject to constraints $g_i(x, y, \ldots) = 0, i = 1 \ldots n$ by introducing new variables, $\lambda_i, i = 1 \ldots n$ and minimizing the new function

$$F(x, y, \ldots) = f(x, y, \ldots) + \sum_{i=1}^{n} \lambda_i g_i(x, y, \ldots)$$

with respect to $x, y, \ldots, \lambda_1, \ldots, \lambda_n$.

For concreteness, let us make point (x, y), initially (1, 0), collinear with line 01, given $(x_0, y_0) = (0, 0)$ and $(x_1, y_1) = (1, 1)$. The error function is $y - x$, which, after linearization, gives the equation $x = y$. This equation is underdetermined, so we will use it as a constraint in minimizing $(x-1)^2 + y^2$. Using the Lagrange multiplier, λ, we must minimize $(x-1)^2 + y^2 + \lambda(y - x)$ with respect to x, y and λ. This gives the matrix equation,

$$\begin{pmatrix} 2 & 0 & -1 \\ 0 & 2 & 1 \\ -1 & 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ \lambda \end{pmatrix} = \begin{pmatrix} 2 \\ 0 \\ 0 \end{pmatrix}$$

This system solves to give $(x, y, \lambda) = (.5, .5, -1)$, and we see that (x, y) has moved perpendicular to the line, minimizing its displacement.

It is worth noting the general case. If we wish to minimize the displacement of all the points in the vector X, subject to the (matrix) constraints $AX = C$, we get the following matrix equation.

$$\begin{pmatrix} 2 & A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} X \\ L^T \end{pmatrix} = \begin{pmatrix} 2X_0 \\ C \end{pmatrix} \tag{2}$$

where L is the vector of λs, X_0 is the vector of original positions, and the superscript ^{T} is the transposition operator.

Overdetermined Equations

Instead of being underdetermined ($< n$ equations in n unknowns) as in the previous section, or just right (n equations in n unknowns), the constraints could be overdetermined ($> n$ equations in n unknowns). In this situation, the extra equations might be redundant but are more likely to conflict, so that no solution is possible. Sutherland uses a transformation of the matrix equation

$$AX = C$$

which minimizes the total squared difference, $(AX - C)^T(AX - C)$. The transformed equation is

$$A^T AX = A^T C \tag{3}$$

Let us call this the *reduction transformation*, since $A^T A$ is an n by n matrix, while A is $> n$ by n.

As an example, let us constrain (x, y), initially (.5,0), to be collinear with both line 01 and line 23, where

$$(x_0, y_0) = (0, 0) \qquad (x_1, y_1) = (1, 0)$$
$$(x_2, y_2) = (0, 1) \qquad (x_3, y_3) = (1, 1)$$

The constraint equations are, with Lagrange multipliers,

$$\begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 1 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ \lambda 1 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$$

$$\begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 1 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ \lambda 2 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}$$

Putting the two equations together gives a system of six equations in four unknowns, which becomes the following 4 by 4 system under the reduction transformation.

$$\begin{pmatrix} 8 & 0 & 0 & 0 \\ 0 & 10 & 2 & 2 \\ 0 & 2 & 1 & 0 \\ 0 & 2 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ \lambda 1 \\ \lambda 2 \end{pmatrix} = \begin{pmatrix} 4 \\ 1 \\ 0 \\ 0 \end{pmatrix}$$

The result is the correct answer: $(x, y) = (0.5, 0.5)$, moving (x, y) the minimum distance to half-way between the two lines.

(Actually, since we are not interested in the values of the λs, and since their equations are symmetrical, they can be merged into one, giving a 3 by 3 system.)

Relaxation and One-Pass Implementations

Consider a more complicated set of constraints.

$$\text{error1}(x_1, y_1) = \begin{vmatrix} x_1 & y_1 & 1 \\ x_0 & y_0 & 1 \\ 0 & 1 & 1 \end{vmatrix}$$

$$\text{error2}(x_1, y_1) = y_1 - x_1$$

$$\text{error1}(x_2, y_2) = \begin{vmatrix} x_2 & y_2 & 1 \\ x_0 & y_0 & 1 \\ 0 & 1 & 1 \end{vmatrix}$$

$$\text{error2}(x_2, y_2) = y_2 + x_2$$

$$\text{error1}(x, y) = \begin{vmatrix} x & y & 1 \\ x_1 & y_1 & 1 \\ 1 & \text{-1} & 1 \end{vmatrix}$$

$$\text{error2}(x, y) = \begin{vmatrix} x & y & 1 \\ x_2 & y_2 & 1 \\ \text{-1} & \text{-1} & 1 \end{vmatrix}$$

(These represent the linkage in [3] that draws a conic section, consisting of a driving lever hanging from point (0, 1) and terminating at (x_0, y_0), and the fixed lines, $y = x$, from point (-1, -1), and $y = -x$, from point (1, -1). (x_1, y_1) is the intersection of the lever with the first line, (x_2, y_2) is the intersection with the second line, and (x, y) traces out the parabola, $y = -x^2$.)

The *relaxation* process consists of choosing one variable, say (x, y), and re-evaluating it to reduce the total error introduced by all the constraints, choosing another variable, and repeating. This process may iterate for some time, even with linear constraints, as in the above example. For example, suppose (x_1, y_1) is initially (-1/2, -1/2) and (x_2, y_2) is initially (-1/4, 1/4), and the lever, (x_0, y_0), is moved to (-1/2, -1). The new position of (x, y) is given by error1(x, y) and error2(x, y) as (-2/3, -4/9). From this, the new position of (x_1, y_1) is given by error1(x_1, y_1), error2(x_1, y_1) and error1(x, y). These overdetermine the solution, so we use equation 3 to obtain (-0.3945, -0.5347). Similarly, (x_2, y_2) becomes (-0.2029, 0.3043).

We then must do further iterations, because these are not the final solutions, as we see next.

The *one-pass* method, when it is applicable, finds an order on the above variables such that a single pass is sufficient. For the linkage problem, it is applicable: evaluate (x_1, y_1) and (x_2, y_2) first, then (x, y). With the placement of (x_0, y_0) at (-1/2, -1), as above, solving the equations in this order gives immediately the correct solution, (x_1, y_1) = (-1/3, -1/3), (x_2, y_2) = (-1/5, 1/5), and (x, y) = (-1/2, -1/4)[3].

For a linear problem such as the conic section linkage above, it may seem appropriate to solve all the linear equations simultaneously. The one-pass method may improve the solution speed, but a clever linear equation package might exploit the ordering automatically. However, even so, a nonlinear problem will require several iterations for each point, and resolving them in order one by one will clearly be faster.

[3] In general, for this configuration of the conic section linkage,

$$(x_1, y_1) = \frac{a}{1+a}(1, 1)$$

$$(x_2, y_2) = \frac{a}{1-a}(1, -1)$$

$$(x, y) = 2a(1, -2a)$$

where a is the slope of the right bisector of the lever.

The conic section "linkage" has all collinearity constraints, and would be hard to build physically. More conventional linkages have constant length constraints, which are quadratic. Here is a Peaucellier linkage (1864), which traces the straight line $x = 3$ from $y = -\sqrt{\frac{6-3\sqrt{2}}{1+\sqrt{2}}}$ to $y = \sqrt{\frac{6-3\sqrt{2}}{1+\sqrt{2}}} = 0.853$. The constraints are

$$(x_0 - 1)^2 + y_0^2 = 1$$

$$x_i^2 + y_i^2 = 13/2$$

$$(x_i - x_0)^2 + (y_i - y_0)^2 = 1/2$$

$$(x - x_i)^2 + (y - y_i)^2 = 1/2$$

where $i = 1, 2$; the lever, (x_0, y_0) moves in a circle of radius 1 about (0, 1); the points (x_i, y_i) move in a circle of radius $\sqrt{13/2}$ about the origin; and (x, y) forms the fourth vertex of an equal-sided parallelogram and traces the straight line[4]. The order of the points for the one-pass method is, as for the conic section machine, (x_0, y_0), then (x_i, y_i) in either order, then (x, y).

For the hexagon problem, a single pass is not possible, and relaxation is needed. This is also a quadratic constraint problem, and so requires iterations for each point, which can be incorporated into the iterations for the relaxation process.

Constraints in the Sketchpad Library

A library of seventeen constraints is described in [3]. Here is a list of thirteen of them. The last one is used to display forces in beams, and the change displayed is since a toggle switch was set.

Name		*Arguments*			*Description*
	1	2	3	4	
collinear	point	point	point		all collinear (no order)
cocircular	point	point	point		1 is centre, 2, 3 on circumference
erect	text				text is erect or on side
horiz/vert	point	point			1 is directly above/below/left/right 2
align	text	point	point		1 is "parallel" to line 2..3
equal length	point	point	point	point	1..2 = $r\times$ 3..4, $r = \frac{1}{3}, \frac{1}{2}, 1, 2, 3$
equal size	text	text			sizes: 1 = $r\times$ 2
distance	scalar	point	point		value of 1 = inches from 2 to 3
size	scalar	text			value of 1 = size of 2 in inches
midpoint	point	point	point		1 is midway from 2 to 3
parallel	point	point	point	point	1..2 is parallel to 3..4
near	text	point			1 is near 2, with space for 5 digits
Δ distance	scalar	point	point		value of 1 = change(inches) from 2 to 3

References

[1] R. Courant and D. Hilbert. *Methods of Mathematical Physics.* Interscience Publisher Inc., New York, 1953. translated from Julius Springer, Berlin, 1937.

[2] Hans Rademacher and Otto Toeplitz. *The Enjoyment of Mathematics.* Princeton University Oress, 1957.

[3] I. E. Sutherland. *Sketchpad A Man-Machine Graphical Communication System.* Garland Publishing, Inc., New York & London, 1980. Ph.D. Thesis, M.I.T., Jan, 1963.

[4]This line is an *inversion* of the circle traced by the lever with respect to the circle of radius $\sqrt{6}$ about the origin. Finding a linkage to trace a straight line was a famous and practical problem in mechanical engineering and mathematics [2].