DOCUMENT RESUME

ED 355 113 SE 053 479

AUTHOR Townsley Kulich, Lisa; Victor, Barbara
TITLE Derive Workshop Matrix Algebra and Linear Algebra.
PUB DATE Nov 92
NOTE 30p.; Paper presented at the International Conference on Technology in College Mathematics (November 13, 1992).
PUB TYPE Guides - Classroom Use - Teaching Guides (For Teacher) (052) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC02 Plus Postage.
DESCRIPTORS Algebra; *Computer Assisted Instruction; Computer Uses in Education; Course Content; Curriculum Guides; Higher Education; Inservice Teacher Education; *Linear Programing; Mathematics Education; Mathematics Instruction; Matrices; *Microcomputers; *Problem Sets; Problem Solving; *Teacher Workshops; Teaching Methods
IDENTIFIERS *DERIVE Computer Algebra System; Linear Algebra; Optimization

ABSTRACT

This document presents the course content for a workshop that integrates the use of the computer algebra system Derive with topics in matrix and linear algebra. The first section is a guide to using Derive that provides information on how to write algebraic expressions, make graphs, save files, edit, define functions, differentiate expressions, calculate both definite and indefinite integrals, calculate Riemann sums, plot polar graphs, and print. Subsequent sections discuss the following topics and their development in Derive: (1) matrix algebra; (2) exercises in systems of linear equations; (3) exercises in linear algebra; (4) solutions to maximization problems in linear programming; (5) graphing strategies in linear programming with Derive; (6) examples and exercises with graphing; (7) the Simplex Method of solving linear programming problems with Derive; (8) examples and exercises with the Simplex Method; (9) linear programming and the Simplex Method in Derive; and (10) a Derive stack for LINPRO.MTH. An additional resource list with 16 references is provided. (MDH)

# ICTCM 11/13/92

# DERIVE WORKSHOP

# MATRIX ALGEBRA

# AND

# LINEAR ALGEBRA

Dr. Lisa Townsley Kulich

Mrs. Barbara Victor

Illinois Benedictine College
5700 College Road
Lisle, Illinois 60532

The Derive Workshop: Matrix Algebra and Linear Algebra was a workshop where participants gained knowledge of the Derive software with specific applications to matrix and linear algebra.

The Derive software is a mathematical assistant for personal computers. It is available for purchase from

SOFT WAREHOUSE
3660 WAIALAE AVENUE
SUITE 304
HONOLULU, HAWAII, USA 96816-3236

PHONE: (808) 734-5810
FAX: (808) 735-1105

Derive software is a menu driven computer algebra system that simplifies, solves and plots mathematical expressions. The system is easy to learn and provides the user with a tool for working with mathematics on an elementary or advanced level. It will perform symbolic operations in algebra, trigonometry, calculus, vectors and matrices. The graphing package allows for 2 or 3 dimension plots.

The software also contains many utility files which are used for specialized topics such as Bessel functions, Orthogonal polynomials and Fesnel integrals.

The user also has the option of creating utility files to deal with a topic requiring mathematical operations. This paper includes such a utility file designed to solve linear programming problems. The utility file is designed to encourage the user to master the techniques necessary to solve a linear programming problem while eliminating the arithmetic.

This utility file on linear programming may be used or shared by anyone who has purchased the Derive software. It is important to note the Derive software is not a shareware and must be purchased.

## TABLE OF CONTENTS


| | |
|---|---|
| Information on Derive | 1 |
| Matrix Algebra | 6 |
| Linear Systems Exercises | 8 |
| Linear Algebra Exercises | 10 |
| Solving Maximization Problems in Linear Programming | 13 |
| Linear Programming with Derive Strategy When Graphing | 14 |
| Graphing - Example and Exercises | 15 |
| Linear Programming with Derive Strategy with the Simplex Method | 17 |
| Simplex Method - Example and Exercises | 18 |
| Linear Programming and the Simplex Method on Derive | 20 |
| Derive Stack for LINPRO.MTH | 23 |
| Resource List | 24 |
| Sunrise/Sunset Charts | 25 |

**Info on Derive**
**10/7/92**

**STARTING AND STOPPING.** To print graphs, call up graphics.com (type "graphics" (RET)) before the command "DERIVE" (RET) to start the program. Hercules graphics cards would require the command "hercules" before startup. When you are finished, Save if desired and "Q" to quit ("Y" to abandon expressions).

**USING DERIVE.**

1. Commands available to you are listed at the bottom of the screen in the COMMAND menu. To select a command, you type the letter that is capitalized in the word. You can also select the highlighted command by hitting the return, or enter key (RET). To move the highlight to the desired command, hit the space bar. WHEN YOU PANIC: touching the escape key (ESC) several times will get you back to this original menu. WHEN YOU ARE DONE: the Quit command will exit Derive after asking you whether you intend to save the expressions you created (see the remark below on saving files).
2. The command Author allows you to create expressions. Hit A (for Author), then type in your expression. After you (RET), the expression is entered on a list, or stack, and has a line number. If you have made a typing error, you can just Author again. Editing commands are discussed later.
3. The command Window allows you to manage your screen space. After selecting this command, you are given other commands. Next, Designate, Close, and Split are the subcommands we will use most. W Next moves you to the next window, as will the F1 button. To create another window, use W Split. You are then prompted as to whether you want to split Horizontally or Vertically. Select this with the appropriate capital letter, and type in the line number at which you wish to split (in Text mode, the screen is 80 x 20). the window in you are currently active is highlighted. To toggle between windows, use F1 key or W Next. To close a window, be sure you are active in that window (toggle there) and use W Close.
4. There is a parenthetical remark about the Help system on DERIVE. If you are using a networked version, only one user can select the command Help at any given time. If you try for Help and get a message that this is not possible, simply abort (type a) and continue with DERIVE. Try for the Help command later.

**ALGEBRA.** To Author algebraic expressions, you need to know what symbols to type. Here are some algebraic symbols and their meaning:

1. $-z$ minus $z$
2. $z + w$ $z$ plus $w$
3. $z - w$ $z$ minus $w$
4. $z * w$ or $zw$ $z$ times $w$
5. $z/w$ $z$ divided by $w$

6. $z \wedge w$ — $z$ raised to the power $W$
7. $z\%$ — $z$ percent $= z/100$
8. $z!$ — $z$ factorial

Remember that the order of operations is parentheses, library functions, exponentiation, multiplication and division, and addition and subtraction, read left to right. Your results will be different if you Author $2 \wedge 3 + 5/4$ or $2 \wedge (3 + 5)/4$ or $2 \wedge (3 + 5/4)$ etc. (see p.20 of manual)

**GRAPHS.** To graph an algebraic expression, it must be a function of one variable. This is the independent variable (the inputs) and will be represented on the horizontal axis. The output will be represented on the vertical axis. An algebraic expression must be authored in an algebraic window, and a graph must be plotted in a 2–D graphics window. The command Plot will automatically cause DERIVE to split the screen, with the option of the graph exhibited Beside, Under, or Overlaying (like a window shade) the algebra stack. Choose B, U, O as you desire, then DERIVE will ask you at which column (row) number you would like the window to split. The given default is a halfway. See item 1 below to proceed from here.

To split the screen yourself before Plotting, look above at how to split a window (USING DERIVE, item 3), then toggle (F1) to the window you would like to graph in. Use the command Window Designate to choose the 2–D designation of that window. You will have to abandon any algebraic expressions in that window, so make sure they are listed elsewhere.

1. Graphic capability: when you start up Derive, you are in Text mode, which will only produce dot graphs. For something nicer, you need to move to Graphics mode. Select Options, then select Display. You then have several items to select.
   i. Now you need Graphics, Medium, Large, Extended, CGA. Do this by typing the capital letters of these options. Then when you cycle back to Graphics, RET. Your screen image will become larger. Some people prefer GHSEC, which keeps a smaller screen (more graph) but becomes monochrome. You may produce a printout of these graphs, see the section on printing.
   ii. Anytime you want to do lots of calculus or matrix algebra, I suggest going back to Text mode (easier to read). Options Display Text High Large Extended RET. The command F5 (gray key) will toggle you between the two most recently chosen modes, usually text (what DERIVE starts in) and graphics mode.
2. Plotting: If you are in the graphing window and choose the command Plot, Derive will plot whatever you have highlighted in the algebra window. If you are in the algebra window, you can use the arrow keys to move the highlight to the expression which you wish to plot. You must then choose the Plot command twice: the first P will take you to the plotting window and the second P will plot the expression.
3. Removing excess plots: In the plotting window, the command Delete will allow you to remove some of your graphs.
4. Scales, Cross hairs, and Zooming: The scale of a graph is noted on the bottom of a graphic screen, or can be found by pressing the Scale command (ESC after you have read the scale). The scale indicates the number of units per tick mark on the graph and may differ in the x– and y–coordinates. The cross hair is a cross on the graph; its

coordinated are listed on the bottom of the screen. The move the cross, use the arrow keys; HOME takes the cross to the origin. Ctrl right or left arrow moves faster right or left, similarly with Pg Up and Pg Dn. MOVE moves the cross hair to coordinates you choose. CENTER centers the graph at the cross hair. To zoom out, you can press F10 and F9 to zoom in. See what happens to the scale as you do that. The Zoom command also works.

**SAVING YOUR FILES.** It would be nice to save some of the work you have generated in a lab session, so you can pick up where you left off the next day.

1. You need a formatted floppy disk, 5 1/4 " or 3 1/2 ". Insert the disk in drive B (or available drive) of the PC. After you have generated all the material you wish to save, select the command Transfer and then the command Save. You will be asked what kind of file (DERIVE), and the name of the file. Yes, it is true that DERIVE will only save lines from the stack, and not graphs or extra windows. Still, that's better than nothing!! Name the file B:*name*.mth, where *name* is what you want to call it. The file specification .mth identifies this as a file that DERIVE can read.
2. To use a file again, after you enter DERIVE, use the command Transfer Load and identify the name of the file (the "B:" identifies the file as coming off your floppy). DERIVE will enter all the old lines of your stack. If you want to see some particular line from before, try the command Jump. It will jump to the line number that you specify. After working on your file again, you may of course save it again. The command Transfer Merge will ***append*** the named file to the end of an existing stack.

**EDITING COMMANDS.**

1. While you are Authoring an expression, you can edit. The Backspace key (so labeled or gray, with a large arrow pointing left) will delete the element to the left of the cursor. The Delete key (Del) will delete the character under the cursor. To move the cursor, use the Ctrl key simultaneously with the s or d key (one goes right, the other left). If you type over previous characters, they will be destroyed. To insert a character, you must be in insert mode–do this by pressing the Ins key. After inserting, go back to overwrite mode by pressing the Ins key again. Notice that the status line will indicate whenever you are in insert mode. To Author an expression that includes previously existing material, try the F3 and F4 keys. While in Author, the F3 key will place at the cursor the material that is highlighted on the stack, F4 puts it in parentheses. Notice that insert versus overwrite mode will make a difference. To move the highlight in the stack, use the arrow keys. The right and left arrows move the highlight to different elements within an expression in the stack. These editing commands can be found in the manual on p. 11 and p. 23.
2. Some students have already asked how to manage the stack better, getting rid of mistakes, etc. You can Remove lines from your stack, with the Remove command. You have to enter what line to begin the removal, TAB to the other field, and enter what line to end the removal. RET, and those lines will vanish from the stack. In addition, the Jump command will jump the highlight to the line number you specify.

**FUNCTIONS.** To define a function like $f(x) = \sin(x)$, you Author $f(x) := \sin(x)$. The ":=" tells DERIVE that you have assigned a functional name. The option Manage Substitute will sub in to any line number a value for the variable. Simplify will give a rational expression for the calculation, and approXimate (X, not A) will give a decimal approximation for the calculation. To plot a function, highlight or Author just the expression, not the "$f(x)$ :="part. You have to Transfer Clear to eliminate $f(x)$, even if removed from the stack, and you will lose your entire stack, so new functions should have different names.

**DIFFERENTIATION.** To differentiatie an expression, use the option Calculus Differentiate. You are asked all the important questions, such as what is to be differentiated, what is the variable, and what order derivative to find. After you RET, the notation is given, eg. $\frac{d}{dx}\sin x$ on the stack, and Simplify will perform the Calculus.

**INTEGRATION.** You can get DERIVE to calculate both indefinite and definite integrals. Use the Calculus Integrate option, and either bring down the expression to be integrated from the stack, or backspace and delete to eliminate the line number, then type in your own expression (RET). The prompt will ask you for the variable of integration; usually this is $x$. DERIVE will convert this to $dx$ that you see. After (RET), you need to select limits of integration; first the lower limit $a$ (some real number) then (TAB) to type in the upper limit $b$. If you are working on an indefinite integral, just hit (RET) without typing in any limits (don't type 0–that is a real limit). The line that appears in the stack should look like integrals in the text.

ex. Calculus, Integrate, backspace/delete the line number, $\sin x$ (RET), $x$ , 0 (TAB) (alt) p (RET) will look on the stack like: $\int_0^\pi \sin x \quad dx$

ex. Calculus, Integrate, backspace/delete the line number, $\sin x$ (RET), $x$ (RET),(RET) will look on the stack like: $\int \sin x \quad dx$

You can ask DERIVE to Simplify either of these, although approXimate is a better choice for a definite integral. For an indefinite integral, DERIVE doesn't add $+C$ to the end–you have to understand that it is there.

**SUMMATION.** To calculate Riemann sums for large n, you use the commands Calculus Sum. Once again, you have many ways to enter the summand. You can create it on a line, then use that line number (RET). You could backspace/delete the line number at the summation prompt and type in the expression (RET). Finally, you could backspace/delete the line number, bring down a highlighted expression with the F3 key, then type in any additional pieces (RET). DERIVE will prompt you for an index (the "ticker") and suggest a possibility–type in whatever is correct (RET). Finally, you need the range of the index: type in the start and (TAB) to type in the final (RET). The stack will show the sigma notation $\Sigma$ with your addend. You can Manage Substitute to input real numbers for any of the variables in your summation, BUT DON'T EVER PUT ANYTHING IN FOR THE INDEX–KEEP IT "I"!! Hence, if you are integrating from $a$ to $b$ and using $n$ rectangles, you can Manage Substitute values for $a$,$b$, and $n$. To massage the sum algebraically, ask DERIVE to Simplify. To calculate a numeric value (after you have substituted for $a$,$b$, and $n$) ask DERIVE to approXimate.

ex. To calculate $1^2 + 2^2 + 3^2 + \ldots + n^2$ for various $n$, do Calculus Sum, backspace/delete the line number, type in $i \wedge 2$ (RET), type in $i$ (RET), type in 1 (TAB) $n$ (RET). The stack should look like $\sum_{i=1}^{n} i^2$ .Then Manage Substitute $i$ for $i$ (always!) and 2 for $n$ and Simplify (do you get 5?). Manage Substitute in the sum again–5 for $n$ and Simplify. You can also Simplify the original sum–you get $n(n+1)(2n+1)/6$ which indicates that DERIVE knows the formula for the sum of consecutive squares (do you?).

**POLAR AND PARAMETRIC PLOTS.** Some special plots are polar coordinates and parametric curves.

1. Parametric equations are Authored $[x(t), y(t)]$. When you Plot, you are asked for the range of the parameter $t$, min to max. This is how to plot a vertical line.
2. Polar plots require that you select Options State Polar to change the inputs to angles and radii. You plot functions of the angle; again you have to specify the range, min to max, of the angle. Remember to change to Options State Rectangular to plot $(x, y)$ coordinates again.

**PRINTING.** The command Transfer Print Printer, executed from an Algebra window, will dump the stack to a printer. If the command "graphics" is typed prior to entering DERIVE, a screen dump can be made of graphs. System requirements include CGA to VGA graphics, no Hercules. Then hold down the Shift and Print Screen keys simultaneously to dump the screen to a dot matrix printer. Screen dumps to a laser printer require a driver: such a free program is "HPPS". On a network, additional key sequences may be required.

**VARIABLE TYPE.** You may choose the command Options Input Word, so that DERIVE will recognize $x1$ as a variable (which would represent $x_1$ in problems). Coefficients will precede the variables. Be sure that you go back to Options Input Character before you try something like *FIT* below.

**CREATING MATRICES.**

1. Use the commands Declare Matrix to enter a matrix. (Recall that you type the capital letter of the command you select.) You will be prompted for the dimension of the matrix. Type in the number of rows desired, then TAB to type in the column field, RET. You are then prompted for the entries, proceeding along the rows. Hit RET after each desired entry is typed. When you have completed the process, the matrix will appear on the stack as you expect it to look, identified by its line number.
2. You can enter a matrix using the Author command, a good choice for small matrices, or vectors. Use the square brackets to form a vector, such as [1,2,1,2]. To create a matrix, you Author a vector of vectors. You list each row as a vector, and separate them by commas: [[1,2,1,2],[0,0,0,0]] is the matrix with 1 2 1 2 as the first row, and all zeroes in the second row.
3. If you Author *vector*($x^n$, $n$,1,5), you will get a vector of elements $x^n$, with $n$ ranging from 1 to 5: $[x, x^2, x^3, x^4, x^5]$. You can generate large matrices this way if they have a pattern: try Authoring *vector*($[x, x^2, x^3]$, $x$,1,8). The difference between [ and ( is very important when you type this. There is a difference in the way you operate on vectors versus column or row matrices. It is easier to make column matrices, especially if you want to transpose, etc.
4. If you Author *identity_matrix(3)*, that is what you will see on the stack. Use the command Simplify to see what you expect.

**MANIPULATING the STACK.** If you need to use a matrix again, you can call it up several ways. While using the Author command, you can refer to a matrix by its line number: for example, *det*(#1). You can also bring down the entire matrix: the F3 key will bring any highlighted expression on the stack down to the Author line. F4 will bring it down in parentheses. Experiment with moving the highlight using the arrow keys. If you Author *B*:= and bring down a matrix with F3, then the variable *B* is assigned to the matrix in all subsequent work and may be referred to as such.

**MATRIX OPERATORS.** Derive recognizes the following matrix operations. Remember to use line numbers for matrix names, and after you have Authored such operations, the command Simplify will execute.

1. addition is +, scalar multiplication is 2A or 2*A, cross product is *cross*($a$, $b$).
2. scalar product of vectors or product of matrices is the period .
3. transpose is the accent ' (Notice which way it points!)
4. exponentiation, including inverses is typed using the carat (shift 6).
5. *det*(A)
6. *row_reduce*(A,B)
7. *trace*(A)
8. *eigenvalues*(A)

9. *charpoly*$(A, z)$ gives the characteristic polynomial of matrix A in terms of variable z. You could use the command soLve to find the roots, which are the eigenvalues. Eigenvectors can be found via row reduction (see 6) or there is a command in a utility file.
10. *abs*$(V)$ calculates the length of a vector $V$.
11. element$(v, n)$ finds the $n$th component of a vector $v$, and element$(M, j, k)$ finds the $j, k$ entry of matrix $M$. Also, element(M,j) finds the $j$th row of a matrix $M$.

**SOLVING LINEAR EQUATIONS.** There are two ways to solve a system of linear equations:

1. Form a vector (Author, and enclose in square brackets) of the linear equations. For example, Author $[x - 3y + z = 5, \quad x + y - 2z = 4, \quad 2x - 2y - z = 9]$. Then use the command soLve.
2. Form the matrices $A$ and $b$ so that your system of equations is represented by $Ax = b$, then *row_reduce*$(A, b)$. You must interpret your answer.

**BEST FIT.** You can find the polynomial of best (least squares) fit for a given set of data points. Make sure you are in the default variable mode, which is Options Input Character. Author, then Simplify, as in the example below.

*fit*[[$x$, $ax^2+bx+c$],[-1.5,0],[0.5,-2],[1.5,-1.5]] Simplifies to the parabola $x^2/2 - x/2 - \frac{15}{8}$ You can plot the points and the approximation.

## Linear Systems Exercises

**REMARKS.** Linear Systems do not have to be square to be soLved, in either form (1-5 or 6).

1. Solve the following linear system:

$$[x + y = 9, \quad 2x - y = 0]$$

You Author the system as written above, using square brackets, and then soLve. Of course, in class one would plot the 2 lines (Author in $y =$ form, then Plot, Plot) to visualize solutions.

2. See how DERIVE handles messy numbers:

$$[3x - 5y + 11.5z = 161.55, \quad -3x + y + z = 4.7, \quad -9x + 0y - 2z = -29.4]$$

When you soLve, the solution is a rational solution; approXimate gives a decimal approXimation.

3. Infinite solutions:

$$[-x + y + 3.5z = 8, \quad 5x + 2y - 7z = 37, \quad 4x + 4y - 2z = 56]$$

4. Infinite solutions with more than one parameter: DERIVE asks you which variables you wish dependent:

$$[3x - y + 11z - 8w = -7, \quad x + 2y - z + w = 0]$$

5. No solutions:

$$[x + y - z = 2, \quad 2x - y + 3z = -5, \quad -x + 2y - 5z = 2, \quad x + y + z = 5]$$

6. Solve the above systems using row reduction: begin with Declare Matrix. The matrix which you form is the augmented matrix $[A|b]$ for the linear system $A\vec{x} = \vec{b}$. Of course, there is no "bar" in DERIVE; you have to imagine where it would be. For problem 1, you would Declare Matrix which is 2 TAB 3 RET, and list the entries as prompted, proceeding row by row. The result on line 13 (or so) of your stack is:

$$\begin{bmatrix} 1 & 1 & 9 \\ 2 & -1 & 0 \end{bmatrix}$$

Next, you Author *row_reduce*(# 13) (RET), or whatever the line number is of your matrix. Then Simplify (RET) to obtain the reduced row echelon form. Interpretation of your results is required, but little or no back-substitution.

**BEST FIT.** FIT is an operator which finds the best fit (least sum of squares of distances) of a curve to data points. If there are $n$ data points, an exactly fitting polynomial of degree $n-1$ will be found, or best fit of polynomials of lesser degree. *FIT* is Authored as listed in the exercised below, then Simplify or approXimate. The data points can be plotted: move the highlight with the arrow keys up to the *FIT* line, then right arrow to highlight the matrix. When you Plot Plot, DERIVE will beep and say "too many variables". Then DERIVE plots the data points and, unfortunately, the $y = x$ line. Use the command Delete First to remove the $y = x$ line. Return to the Algebra window and highlight, then Plot the curve which represents the Fit.

**REMARK.** Make sure you are in the default variable type which is Options Input Character (RET), so that DERIVE recognizes $ax$ below as the product of two variables $a$ and $x$, not a single variable string.

1. Author $fit[[x, ax + b], [2, -2], [-1, 1.5], [6.2, 3]]$ (RET). Then use the right arrow to highlight just the matrix part of the line and Plot,Plot. Take out the $y = x$ by Delete First, adjust the Scale to $x = 2$ TAB $y = 2$ (RET) to see all the points, and return to Algebra window. Use the up arrow to highlight the entire Fit statement again, Simplify the Fit, and Plot Plot. Compare to the *fit* with just the first two points. When you are finished with your plots, Delete All before returning to the Algebra window.

2. Author $fit[[x, ax^2 + bx + c], [2, -2], [-1, 1.5], [1, 5], [0, 37/6]]$ (RET). Then Plot the points, Algebra, Simplify and plot the fit. When you Plot the parabola, the Scale should be $x = 1$ TAB $y = 2$.

**REMARKS.** DERIVE is using the matrix algebra solution which ·· lves inverting a matrix, so the $y-$coordinates must be a function of $x$.

3. Refer to the graph of sunrise/sunset times on the last page this booklet. Gather some data to create a reasonalble approximating polynomial.

## Linear Algebra Exercises

**REMARKS.** To create matrices, see page 6. The Declare Matrix command is easiest, although you can Author. Every matrix is a column vector of row vectors, and vectors are typed with square brackets, elements separated by commas. The Declare vectoR command is good for Calculus 3, but not for Linear Algebra applications, an $n$ by 1 matrix works better (for example when you transpose).

Use the following matrices and vectors below. Create them with Declare Matrix. Refer to them in calculations by line number (# 14, for example) or you can name them by Authoring $A :=$ and then using the gray F3 key to bring the appropriate matrix down to the author line. Then the matrix can be referred to by variable name.

$$A := \begin{bmatrix} 1 & -1 & 4 \\ 3 & 2 & -1 \\ 2 & 1 & -1 \end{bmatrix} \qquad B := \begin{bmatrix} 3 & .5 & .45 \\ -4 & 1.5 & -1 \\ 0 & 0 & -1 \end{bmatrix} \qquad C := \begin{bmatrix} 2 & \pi & \sqrt{2} \\ 1 & -.5 & 2/3 \end{bmatrix}$$

Use *pi* for $\pi$ and *sqrt 2* for $\sqrt{2}$.

$$\vec{v} := [\,-2 \quad -3 \quad 1\,] \qquad \vec{u} := \begin{bmatrix} -1 \\ 4 \\ 1/2 \end{bmatrix}$$

1. Author and Simplify $A + 4B$ .

2. Dot and matrix products are typed with a period. $C.A$ and $A.B$ . Compare approX-imate to Simplify.

3. $C.\vec{u}$ , $\vec{u}.\vec{v}$, $\vec{v}.\vec{u}$ .

4. Powers of square matrices formed by exponentiation: find $A^{10}$. The carat $\wedge$ is used for exponentiation, the shift 6 key.

5. Author $A.[[x],[y],[z]] = \vec{u}$ , Simplify and soLve. Notice that you retutn to linear systems!

6. Check $det(A)$ to verify that $A$ is invertible, then Author and Simplify $A^{-1}$.

7. Another way to solve problem 5 is to multiply the equation by the matrix representing $A^{-1}$. To Author this, highlight the matrix that is $A^{-1}$, use F3 to bring it down without retyping, then type the dot product . and move the highlight up to the unsimplified

equation of problem 5. F4 will bring it down in parentheses. RET and Simplify. (Remind students to check determinant first; unique solutions exist if and only if the determinant is nonzero.)

8. The transpose is the "infix" operator, it is the left accent ' key. Compare $(A^{-1})'$ to $(A')^{-1}$. Also transpose the vector $\vec{v}$. Note that if you have commas, the vector doesn't transpose. For this reason, I always create vectors for linear algebra by Declare Matrix.

9. The cross product is *cross*. Here, since this is a calculus operation, you need vectors with commas! Find $cross([-2,-3,1],[-1,4,1/2])$.

10. Left multiply $A$ by 2*identity_matrix*(3) and Simplify. Experiment by Authoring other elementary matrices (the identity matrix changed by a single row operation) and multiplying them to perform row reduction.

11. Length of a vector: Author and Simplify $|\vec{v}|$.

12. To see if the vector $\vec{u}$ is in the span of the columns of $A$, Author *row_reduce* $(A,\vec{u})$. You can also create a new matrix, the augmented matrix, instead (see problem 6, page 8), as *row_reduce* works with either one or two arguments. The result is more than yes or no, you also get the coefficients!

13. Similarly, you can use *row_reduce* to determine whether a set of vectors is dependent, to find the dimension of a set of vectors, and to find a change of basis matrix for 2 bases such as the columns of $A$ and $B$.

14. Define a linear transformation $T$: here the vectors need commas! Use Declare Function, name it $T$, give it a Value $[2x-3y,-2z,x+2y]$ . Then Author and Simplify $T([1,0,0])$, $T([0,1,0])$, and $T([0,0,1])$. Create the matrix $T1$ which has columns the image of the 3 canonical basis vectors. Verify that $T([2,-9,1.5])$ yields the same value as the product of $T1$ and the vector $[2-91.5]'$. (Watch the commas!)

15. The characteristic polynomial is found by Author and Simplify *charpoly*$(A)$. Then soLve the polynomial to find its roots, the eigenvalues. You can also Author and Simplify eigenvalues(A). Find bases for eigenspaces by *row_reduce* of the appropriate augmented matrix. By loading the utility file "vector.mth", you can find *exact_eigenvector*(A,1) , a basis for the eigenspace of the matrix $A$ for the eigenvalue 1 (or whatever $\lambda$). *Approx_eigenvector* is appropriate for large numeric matrices, yielding an approximate eigenvector of length 1.

16. The applications of Markov chains work nicely on DERIVE. Recall that a Markov

chain is represented by a transition matrix $M$ whose $ij$ entry represents the probability of transition from state $i$ to state $j$. Regular transition matrices (some power of $M$ contains all positive entries) have unique fixed vectors. It is easy to approXimate large powers of $M$ on DERIVE, and the rows of $M^n$ will converge to the fixed vector. One can also Declare Variable $s$ with value $[x, y, z]$, and Author $s.M = s$. Then Simplify, include the equation $x + y + z = 1$, and soLve, approX. For example, consider weather prediction. If the weather changes tomorrow to good, indifferent, or bad depending on whether it is good, indifferent, or bad today, this can be represented by a transition matrix, say

$$M := \begin{bmatrix} & G & I & B \\ G & .5 & .3 & .2 \\ I & .3 & .4 & .3 \\ B & .2 & .4 & .4 \end{bmatrix}$$

Decide what the long term weather probabilities are. (Don't use G,I,B in your matrix, of course.)

**Solving Maximization Problems in Linear Programming**

The business majors at Illinois Benedictine College take a two semester course in finite mathematics[1] which includes an introduction to linear programming. The topic includes graphing linear inequalities in two variables, solving linear programming problems graphically and solving linear programming problems by the Simplex Method.

Derive is an excellent tool for the graphical solution of systems of linear inequalities. The student quickly grasps the concept and has a good understanding of the region of feasible solutions and the objective function.

When it comes to solving linear programming problems by the Simplex Method there are software packages that will solve the problem. Since our purpose in this course is to teach students how to solve linear programming problems, these packages are not suitable.

This presentation includes a user created utility file for Derive that requires the student to master the strategy necessary to solve linear programming problems by the Simplex Method. The idea is to stress the technique and eliminate the arithmetic.

---

1. Margaret A. Lial, Charles D. Miller and Thomas W. Hungerford, **Mathematics with Applications in the Management, Natural and Social Sciences**, Fifth edition; Harper Collins Publishers; c.1991

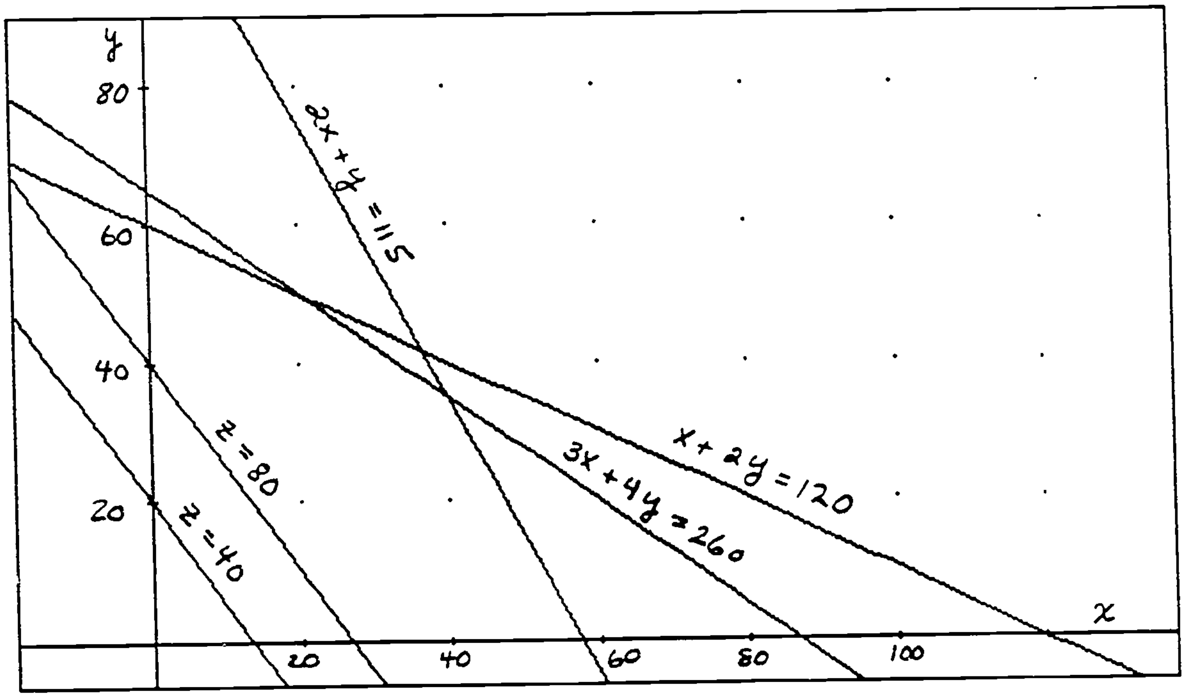
**LINEAR PROGRAMMING WITH DERIVE**

STRATEGY WHEN GRAPHING

- Enter the constraints as equations.
- Solve each equation for y.
- Decide on a feasible scale.
- Prepare the graphing screen to display only the first quadrant.
- Plot the lines.
- Determine which intersections are needed.
- Use Derive [Eq.1, Eq.2] to find the intersections.
- Assign two or three values to Z in the objective function and plot these lines.
- Visually determine where the maximum is.
- Calculate the maximum.

**EXAMPLE Solution by Graphing**

Maximize: $Z = 3x + 2y$
Subject to: $x \geq 0$ , $y \geq 0$
$x + 2y \leq 120$
$3x + 4y \leq 260$
$2x + y \leq 115$
Solution: $z = 190$ when $x = 40$ and $y = 35$.




Scale x=20 and y=20 z=3x+2y

**EXERCISES**

Maximize: $z = 4x + 3y$

Subject to: $x \geq 0$ , $y \geq 0$
$y \leq 25$
$2x + 5y \leq 155$
$3x + 2y \leq 150$

Solution $z = 205$ when $x = 40$ $y = 15$

Maximize: $z = 2x + 2y$

Subject to: $x \geq 0$ , $y \geq 0$
$-x + 5y \leq 625$
$x + y \leq 275$
$3x - y \leq 625$

Solution $z = 550$ when $125 \leq x \leq 225$ on the line $x + y = 275$

Maximize: $z = 2x + 3y$

Subject to: $x \geq 0$ , $y \geq 0$
$x + 5y \leq 30$
$3x + 7y \leq 46$
$x + y \leq 10$
$2x + y \leq 18$

Solution $z = 24$ when $x = 6$ $y = 4$

**LINEAR PROGRAMMING WITH DERIVE**

STRATEGY WITH THE SIMPLEX METHOD

- Prepare the utility file.
  **Transfer, Load,** *linpro.mth*

- Enter constraints and objective function.
  **Declare, Matrix**
  Rows = # of constraints plus 1.
  Columns = # of variables plus 1.

- Create initial simplex plateau.
  ***Slack_matrix(v,i,j)***

- Select the pivot element.
  Most negative entry in indicator row.
  Smallest ratio,constant/coefficient.
  ***Divide_elements(v,i)***

- Perform the pivot operation.
  ***Pivot(v,i,j)***

- Read the solution or repeat the process.

- Solution found when indicator row contains only non-negative entries.

- To facilitate reading solution.
  ***Divide_row(v,i,j)***

**EXAMPLE Solution by Simplex**

Maximize: $Z = 3x_1 + 2x_2$

Subject to: $x_1 \geq 0$ , $x_2 \geq 0$
$x_1 + 2x_2 \leq 120$
$3x_1 + 4x_2 \leq 260$
$2x_1 + x_2 \leq 115$
Solution: $z = 190$ when $x_1 = 40$, $x_2 = 35$, $x_3 = 10$, $x_4 = 0$, $x_5 = 0$

$$\begin{bmatrix} 1 & 2 & 1 & 0 & 0 & 0 & 120 \\ 3 & 4 & 0 & 1 & 0 & 0 & 260 \\ 2 & 1 & 0 & 0 & 1 & 0 & 115 \\ -3 & -2 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

## EXERCISES

Maximize: $Z = 5x_1 + 3x_2 + 2x_3$

Subject to: $x_1 \geq 0$ , $x_2 \geq 0$ $x_3 \geq 0$
$x_1 + 3x_2 + 7x_3 \leq 15$
$2x_1 + x_2 + 5x_3 \leq 12$

Solution: $z = \frac{159}{5}$ when $x_1 = \frac{21}{5}$, $x_2 = \frac{18}{5}$, $x_3 = 0$, $x_4 = 0$ and $x_5 = 0$.

Maximize: $Z = 3x_1 + x_2 + 2x_3$

Subject to: $x_1 \geq 0$ , $x_2 \geq 0$ $x_3 \geq 0$
$x_1 + 2x_2 + x_3 \leq 80$
$3x_1 + 6x_2 + 2x_3 \leq 20$
$3x_1 + x_2 + 3x_3 \leq 60$

Solution: $z = 20$ when $x_1 = \frac{20}{3}$, $x_2 = 0$, $x_3 = 0$, $x_4 = \frac{220}{3}$, $x_5 = 0$ and $x_6 = 40$.

Maximize: $Z = 2x_1 + 3x_2 + 5x_3 + 3x_4$

Subject to: $x_1 \geq 0$ , $x_2 \geq 0$ $x_3 \geq 0$ $x_4 \geq 0$
$4x_1 + 2x_2 + 3x_3 + 2x_4 \leq 12$
$2x_1 + 4x_2 + x_3 + x_4 \leq 16$
$8x_1 + 3x_2 + 6x_3 + x_4 \leq 24$
$x_1 + x_2 + 2x_3 + 4x_4 \leq 32$
Solution: $z = 20$ when $x_1 = 0$, $x_2 = 0$, $x_3 = 4$, $x_4 = 0$, $x_5 = 0$, $x_6 = 12$, $x_7 = 0$ and $x_8 = 24$.

**LINEAR PROGRAMMING AND THE SIMPLEX METHOD ON DERIVE**

Derive can be used to solve linear programming problems by the simplex method. The commands included in this utility file are designed to reinforce the techniques necessary to obtain the solution manually. The utility file contains commands that will solve maximization or minimization problems where there are no mixed constraints.

To begin, enter the system of inequalities as a matrix without the slack variables (Use **D**eclare **M**atrix). To prepare for a maximization problem enter the last (indicator) row with its minus signs. To prepare for a minimization problem enter minus signs in the last (constant) column and use Derive's transpose command before creating the basic simplex plateau.

There are six Derive commands in this utility file. The user must be familiar with four of them. The other two are necessary to create the basic simplex tableau.

1. **Create the basic simplex plateau.**
   Use the command slack_matrix(v,r,c), where v identifies the matrix representing the original problem, r is the number of rows and c the number of columns in the original matrix.

```
          ****************************************
SLACK_MATRIX(v,r,c):=MIX_ELEMENTS(FILL_MATRIX(v,r),r,c)`
          ****************************************
```

   The slack_matrix command uses two commands, fill_matrix and mix_elements. The fill_matrix command attaches an identity matrix of the appropriate size to the original matrix and mix_elements places the identity matrix between the coefficients and the constants to create the basic simplex tableau.

```
          ****************************************
FILL_MATRIX(v,r):=APPEND(v`,IDENTITY_MATRIX(r))

MIX_ELEMENTS(v,r,c):=VECTOR(IF(m_<c,ELEMENT(v,m_),
     IF(m_=c+r,ELEMENT(v,c),ELEMENT(v,m_+1))),m_,r+c)
          ****************************************
```

**2. Locate the pivot**

To find the pivot position, the constant in each row must be divided by the entry in the column that corresponds with the most negative number in the indicator row (last row). Use the command divide_elements(v,c), where v identifies the matrix and c is the number of the column targeted as the pivot. The divide_elements command divides the entire row by the entry in column c when that entry is greater than zero. The quotients will appear in the constants column and the column targeted as the pivot will contain a one if it was positive or its previous entry otherwise. The smallest positive quotient with a one in the pivot column directs the user to the pivot.

```
          ****************************************
DIVIDE_ELEMENTS(v,c):=VECTOR(IF(m_=DIMENSION(v),ELEMENT(v,m_),
     IF(ELEMENT(v,m_,c)<=0, ELEMENT(v,m_),
     ELEMENT(v,m_)/ELEMENT(v,m_,c))),m_,DIMENSION(v))
          ****************************************
```

**3. Use the pivot command**

At this point the pivot position should contain a one. To place zeroes above and below use pivot(v,i,j), where v identifies the matrix, i is the row and j the column of the pivot location.

```
          ****************************************
PIVOT(v,i,j):=VECTOR(IF(m_=i,ELEMENT(v,m_),
     ELEMENT(v,m_)-ELEMENT(v,m_,j)/ELEMENT(v,i,j)*ELEMENT(v,i)),
     m_,DIMENSION(v))
          ****************************************
```

**4. Read a solution**

It may be necessary to divide a row by a constant to read a solution. In order to do this use the command divide_row(v,i,d) where v identifies the matrix, i is the row to be divided and d is the divisor.

```
          ****************************************
DIVIDE_ROW(v,i,d):=VECTOR(IF(m_=i,ELEMENT(v,m_)/d,
     ELEMENT(v,m_)),m_,DIMENSION(v))
          ****************************************
```

**5. Adjustments for problems with mixed constraints**

When mixed constraints occur, it is necessary to insert a minus sign to replace the slack variable with a surplus variable. The command chs(v,i,j) changes the sign of the element (i,j) in matrix v. The command scale_element(v,i,s) is a Derive command from the vector utility file that multiplies a vector by a scalar.

```
            ****************************************
CHS(v,i,j):=VECTOR(IF(m_/=i,ELEMENT(v,m_),
  ELEMENT(v,m_) . SCALE_ELEMENT(IDENTITY_MATRIX(DIMENSION(v')),
  j,-1)),m_,DIMENSION(v))
            ****************************************

            ****************************************
SCALE_ELEMENT(v,i,s):=VECTOR(IF(m_=i,
   s*ELEMENT(v,i),ELEMENT(v,m_)),m_,DIMENSION(v))
            ****************************************
```

**DERIVE STACK FOR LINPRO.MTH**

```
1: "Linear programming with Derive on a matrix called v"

2: "MIX_ELEMENTS and FILL_ELEMENTS prepares the slack_matrix"

3: MIX_ELEMENTS(v,r,c):=VECTOR(IF(m_<c,ELEMENT(v,m_),
   IF(m_=c+r,ELEMENT(v,c),ELEMENT(v,m_+1))),m_,r+c)

4: FILL_MATRIX(v,r):=APPEND(v`,IDENTITY_MATRIX(r))

5: "SLACK_MATRIX(v,r,c) puts slack variables in place,
    r= # rows, c= # columns"

6: SLACK_MATRIX(v,r,c):=MIX_ELEMENTS(FILL_MATRIX(v,r),r,c)`

7: "DIVIDE_ELEMENTS(v,c)
    divides each element in row i by the entry in (i,c)"

8: DIVIDE_ELEMENTS(v,c):=VECTOR(IF(m_=DIMENSION(v),
   ELEMENT(v,m_),IF(ELEMENT(v,m_,c)<=0,ELEMENT(v,m_),
   ELEMENT(v,m_)/ELEMENT(v,m_,c))),m_,DIMENSION(v))

9: "PIVOT (v, i, j) Puts zeros above and below entry (i,j)"

10: PIVOT(v,i,j):=VECTOR(IF(m_=i,ELEMENT(v,m_),
    ELEMENT(v,m_)-ELEMENT(v,m_,j)/ELEMENT(v,i,j)*ELEMENT(v,i)),
    m_,DIMENSION(v))

11: "DIVIDE_ROW(v,i,d) divides row i by a number, d"

12: DIVIDE_ROW(v,i,d):=VECTOR(IF(m_=i,ELEMENT(v,m_)/d,
    ELEMENT(v,m_)),m_,DIMENSION(v))

13: "CHS(v,i,j) changes the sign of the element in row i,
     column j."

14: SCALE_ELEMENT(v,i,s):=VECTOR(IF(m_=i,s*ELEMENT(v,i),
    ELEMENT(v,m_)),m_,DIMENSION(v))

15: CHS(v,i,j):=VECTOR(IF(m_/=i,ELEMENT(v,m_),
    ELEMENT(v,m_) . SCALE_ELEMENT(IDENTITY_MATRIX(DIMENSION(v`)),
    j,-1)),m_,DIMENSION(v))
```

## RESOURCE LIST
## CAS and Calculus, Linear Algebra


[1] *Calculus Laboratories with Maple: A tool, not an Oracle*, W. Bauldry and J. Fiedler, Brooks/Cole Publishing, 1991.

[2] *Uses of Technology in the Mathematics Curriculum: A Manual of Laboratory Exercises and Resources for College Teachers*, B. Evans and J. Johnson, Oklahoma State University, 1990.

[3] *Learning by Discovery*, A. Solow, Editor; NSF Grant Associated Colleges of the Midwest and Great Lakes College Association: Preliminary Edition.

[4] *Applications of Calculus*, P. Straffin, Editor; NSF Grant Associated Colleges of the Midwest and Great Lakes College Association: Preliminary Edition.

[5] *Calculus Problems for a New Century*, R. Fraga, Editor, NSF Grant Associated Colleges of the Midwest and Great Lakes College Association: Preliminary Edition.

[6] *Problems for Student Investigation*, M. Jackson, Editor, NSF Grant Associated Colleges of the Midwest and Great Lakes College Association: Preliminary Edition.

[7] *Calculus and the DERIVE Program: Experiments with the Computer*, L. Gilligan and J. Marquardt, Gilmar Publishing, 1990.

[8] *Calculus Applications in Engineering and Science*, S. Goldenberg and H. Greenwald, D.C. Heath, 1990.

[9] *Calculus Laboratories Using DERIVE* L. C. Leinbach, Wadsworth Publishing, 1991.

[10] *Calculus Activities for Graphic Calculators*, D. Pence, PWS Kent Publishing, 1990.

[11] *Graphing Calculator Laboratory Manual for Calculus*, C. Beckmann and T. Sundstrom, Addison Wesley Publishing, 1990.

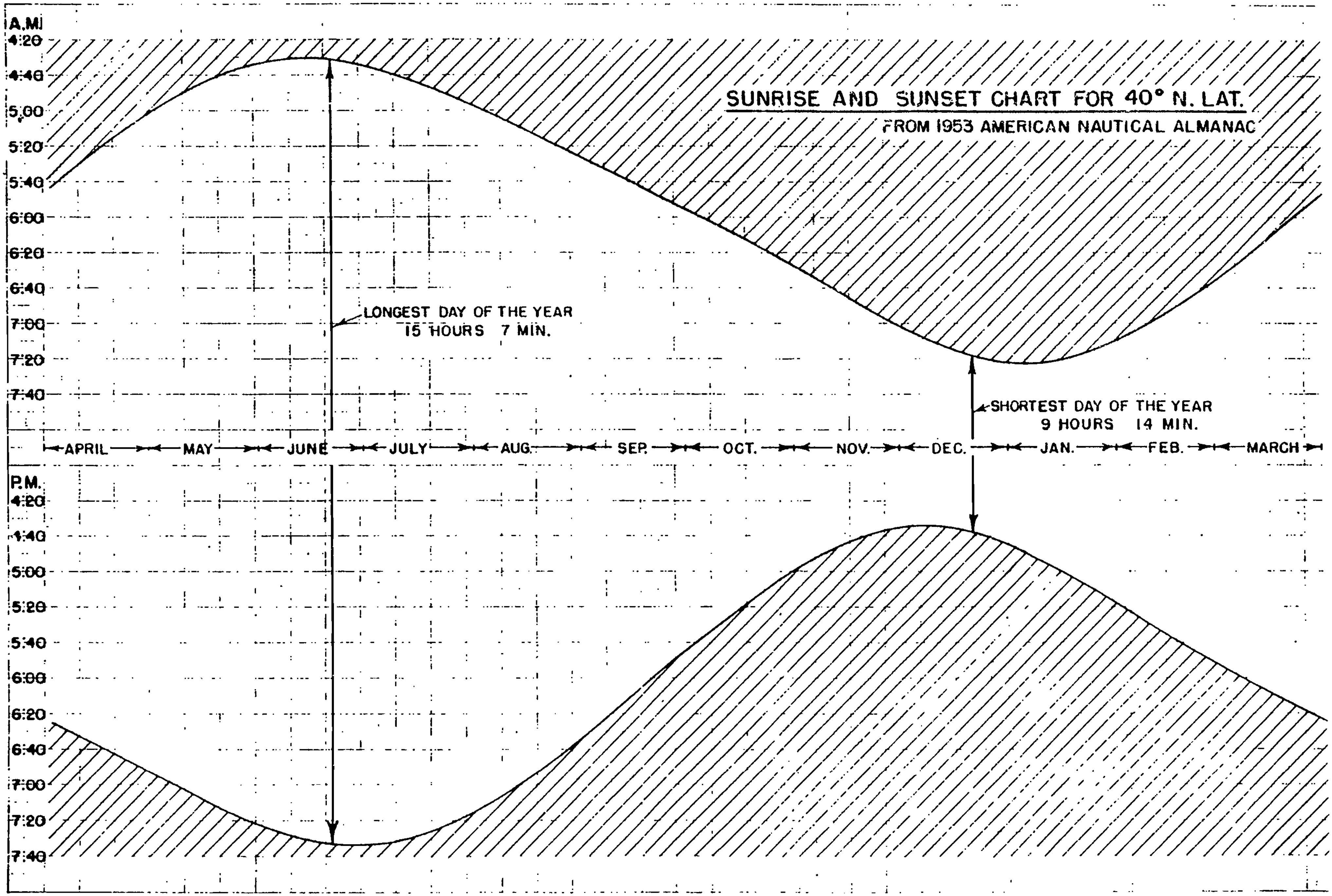
[12] *Proceeding of the International Conferences on Technology in Collegiate Mathematics—Contributed Papers*, (three proceedings), Ohio State University and Addison Wesley Publishing.

[13] *Computer Enhancement of the Calculus*, W. Boyce and E. Luchins, Rensselaer Polytechnic Institute.

[14] *Bringing Calculus to Life*, R. Decker and J. Williams, graphing calculator and CAS versions, Prentice Hall, 1992.

[15] *Discovering Calculus with DERIVE*, J. Johnson and B. Evans, Wiley, 1992.

[16] *Linear Algebra Experiments using the DERIVE Program*, M. Salter and L. Gilligan, Gilmar, 1992.


SUNRISE AND SUNSET CHART FOR 40° N. LAT.
FROM 1953 AMERICAN NAUTICAL ALMANAC
A.M.
4:20
4:40
5:00
5:20
5:40
6:00
6:20
6:40
7:00
7:20
7:40
LONGEST DAY OF THE YEAR
15 HOURS 7 MIN.
SHORTEST DAY OF THE YEAR
9 HOURS 14 MIN.
APRIL
MAY
JUNE
JULY
AUG.
SEP.
OCT.
NOV.
DEC.
JAN.
FEB.
MARCH
P.M.
4:20
4:40
5:00
5:20
5:40
6:00
6:20
6:40
7:00
7:20
7:40