MATLAB

for Engineering Applications

Fifth Edition

William J. Palm III

© McGraw Hill LLC. All rights reserved. No reproduction or distribution without the prior written consent of McGraw Hill LLC.

Chapter 08

Linear Algebraic Equations

© McGraw Hill LLC

Linear Algebraic Equations

Linear algebraic equations such as

$5x - 2y = 13$

$7x + 3y = 24$

occur in many engineering applications. For example, electrical engineers use them to predict the power requirements for circuits; mechanical, and aerospace engineers use them to design structures and machines; and industrial engineers apply them to design schedules and operations.

1. Unique solution
2. No solution: Overdetermined
3. Infinite solution: Underdetermined

© McGraw Hill LLC

Matrix notation enables us to represent multiple equations as a single matrix equation

For example, consider the following set:

$$2x_1 + 9x_2 = 5$$
$$3x_1 - 4x_2 = 7$$

This set can be expressed in vector-matrix form as

$$\begin{bmatrix} 2 & 9 \\ 3 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 5 \\ 7 \end{bmatrix}$$

which can be represented in the following compact form

$$\mathbf{Ax} = \mathbf{b}$$

and the solution can be determined by:

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$$

© McGraw Hill LLC

Matrix notation enables us to represent multiple equations as a single matrix equation

For the equation set $\mathbf{Ax} = \mathbf{b}$,

- if $|\mathbf{A}| = 0,$ then there is no unique solution.
- Depending on the values in the vector **b**, there may be no solution at all, or an infinite number of solutions.

© McGraw Hill LLC

Limitations of Matrix Methods

• The matrix inverse method will warn us if a unique solution does not exist, but it **does not** tell us whether there is no solution or an infinite number of solutions.

• The method is limited to cases where the matrix A is square, that is, cases where the number of equations equals the number of unknowns.

• A method is needed to determine whether an equation set has a solution and whether it is unique. The method requires the concept of the **rank** of a matrix.

© McGraw Hill LLC

Rank of a Matrix

• An m x n matrix A has a rank r≥1 if and only if |A| contains a nonzero r x r determinant and every square sub-determinant with r + 1 or more rows is zero.

For example, the rank $A = \begin{vmatrix} 3 & -4 & 1 \\ 6 & 10 & 2 \\ 9 & -7 & 3 \end{vmatrix} = 0$ is 2 because |A| = 0

while |A| contains at least one nonzero 2 × 2 sub-determinant. If A is n × n, its rank is n if det(A) ≠ 0

© McGraw Hill LLC

Singular Problem

A *singular* problem refers to a set of equations having either no unique solution or no solution at all. For example, the set

$$3x - 4y = 5$$

$$6x - 8y = 10$$

is singular and has no unique solution because the second equation is identical to the first equation, multiplied by 2. The graphs of these two equations are identical. All we can say is that the solution must satisfy $y = (3x - 5)/4$, which describes an infinite number of solutions.

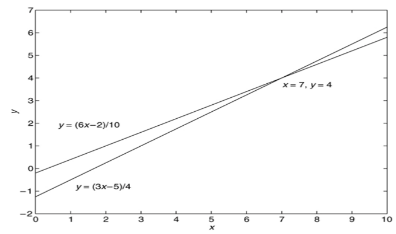
© McGraw Hill LLC

Examples

The equations

$$6x - 10y = 2$$

$$3x - 4y = 5$$

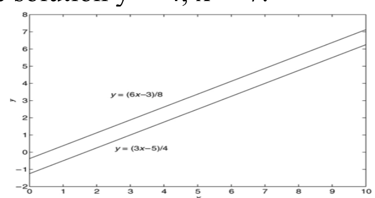


have graphs that intersect at the solution y = 4, x = 7.

On the other hand, the set

$$3x - 4y = 5$$

$$6x - 8y = 3$$



is singular but has no solution. The graphs of these two equations are distinct but *parallel* (see the next slide). Because they do not intersect, no solution exists.

© McGraw Hill LLC

MATLAB `inv` Command

The MATLAB command `inv(A)` computes the inverse of the matrix **A**. The following MATLAB session solves the following equations using MATLAB.

$$2x1 + 9x2 = 5$$

$$3x1 - 4x2 = 7$$

```
>>A = [2,9;3,-4];b = [5;7]
>>x = inv(A)*b
x =
   2.3714
   0.0286
```

If you attempt to solve a singular problem using the `inv` command, MATLAB displays an error message.

© McGraw Hill LLC

Existence and Uniqueness of Solutions

The set $\mathbf{Ax} = \mathbf{b}$ with m equations and n unknowns has solutions if and only if

$$\text{rank}[\mathbf{A}] = \text{rank}[\mathbf{A}\ \mathbf{b}] \qquad (1)$$

Let $r = \text{rank}[\mathbf{A}]$.

- If condition (1) is satisfied and if $r = n$, then the solution is unique.
- If condition (1) is satisfied but $r < n$, an infinite number of solutions exists and r unknown variables can be expressed as linear combinations of the other $n - r$ unknown variables, whose values are arbitrary.

© McGraw Hill LLC

Homogeneous Case

Assume A is an $m \times n$ matrix representing m equations and n unknowns.

The homogeneous set $\mathbf{Ax} = \mathbf{0}$ is a special case in which $\mathbf{b} = \mathbf{0}$.

For this case $\text{rank}[\mathbf{A}] = \text{rank}[\mathbf{A}\ \mathbf{b}]$ always, and thus the set always has the trivial solution $\mathbf{x} = \mathbf{0}$.

A nonzero solution, in which at least one unknown is nonzero, exists if and only if $\text{rank}[\mathbf{A}] < n$.

If $m < n$, the homogeneous set always has a nonzero solution.

© McGraw Hill LLC

Homogeneous Case

If the number of equations equals the number of unknowns and if $|\mathbf{A}| \neq 0$, then the equation set has a solution and it is unique.

If $|\mathbf{A}| = 0$ or if the number of equations does not equal the number of unknowns, then you must use the methods presented in Sections 8.3 or 8.4.

© McGraw Hill LLC

Homogeneous Equations

Consider the following set of *homogeneous equations* (which means that their right sides are all zero)

$$6x + ay = 0$$

$$2x + 4y = 0$$

where a is a parameter. Multiply the second equation by 3 and subtract the result from the first equation to obtain

$$(a - 12)y = 0$$

The solution is $y = 0$ *only if* $a \neq 12$; if $a = 12$, there is an infinite number of solutions for x and y, where $x = -2y$.

© McGraw Hill LLC

Left-Division Method

MATLAB provides the *left-division* method for solving the equation set **Ax** = **b**. The left-division method is based on Gauss elimination. (Section 8.2) To use the left-division method to solve for **x**, type `x = A\b`. For example,

$$\begin{bmatrix} 6 & -10 \\ 3 & -4 \end{bmatrix}\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 2 \\ 5 \end{bmatrix}$$

```
>> A = [6, -10; 3, -4]; b = [2; 5];

>> x = A\b

x =

   7     4
```

This method also works in some cases where the number of unknowns does not equal the number of equations.

© McGraw Hill LLC

Example

solve the following set of linear equations.

$$3\,x_1 + 2x_2 - 9x_3 = -65$$
$$-9x_1 - 5x_2 + 2x_3 = 16$$
$$6x_1 + 7x_2 + 3x_3 = 5$$

$$A = \begin{bmatrix} 3 & 2 & -9 \\ -9 & -5 & 2 \\ 6 & 7 & 3 \end{bmatrix} \qquad b = \begin{bmatrix} -65 \\ 16 \\ 5 \end{bmatrix}$$

© McGraw Hill LLC

Example

solve the following set of linear equations.

$$3\,x_1 + 2x_2 - 9x_3 = -65$$
$$-9x_1 - 5x_2 + 2x_3 = 16$$
$$6x_1 + 7x_2 + 3x_3 = 5$$

$$A = \begin{bmatrix} 3 & 2 & -9 \\ -9 & -5 & 2 \\ 6 & 7 & 3 \end{bmatrix} \quad b = \begin{bmatrix} -65 \\ 16 \\ 5 \end{bmatrix}$$

```
>> A = [3,2,-9;-9,-5,2;6,7,3];
rank(A)
b = [-65;16;5];
x = A\b
ans =  3
```

Because **A** is 3 × 3 and rank(**A**) = 3, which is the number of unknowns, a unique solution exists.

```
x =     2.0000
   -4.0000
    7.0000
```

© McGraw Hill LLC

Underdetermined Systems

An *underdetermined system* does not contain enough information to solve for all of the unknown variables, usually because it has fewer equations than unknowns.

Thus, an infinite number of solutions can exist, with one or more of the unknowns dependent on the remaining unknowns.

© McGraw Hill LLC

Simple Examples of Underdetermined Systems 1

$$x + 3y = 6$$

All we can do is solve for one of the unknowns in terms of the other; for example, $x = 6 - 3y$. An infinite number of solutions satisfy this equation.

When there are more equations than unknowns, the left-division method will give a solution with some of the unknowns set equal to zero. For example,

```
>>A = [1, 3]; b = 6;
>>solution = A\b
solution =
   0
   2
```

which corresponds to $x = 0$ and $y = 2$.

© McGraw Hill LLC

Simple Examples of Underdetermined Systems

An infinite number of solutions might exist even when the number of equations equals the number of unknowns.

This situation can occur when $|\mathbf{A}| = 0$.

For such systems the matrix inverse method and Cramer's method will not work, and the left-division method generates an error message warning us that the matrix **A** is singular.

In such cases the *pseudoinverse method*

```
x = pinv(A)*b
```

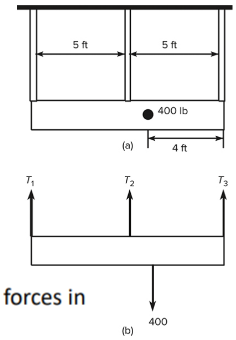
gives one solution, the *minimum norm solution.*

$$pinv(A)=\left(A^t A\right)^{-1} A^t$$

© McGraw Hill LLC

A Statically Indeterminate Problem: A Light Fixture and Its Free-Body Diagram

Determine the forces in the three equally spaced supports that hold up a light fixture. The supports are 5 ft apart. The fixture weighs 400 lb, and its mass center is 4 ft from the right end. Obtain the solution using the MATLAB left-division method and the pseudoinverse method.



T1 + T2 + T3 - 400 = 0
400(4) - 10T1 - 5T2 = 0

Total moment at the right point

T1 , T2 T3 are the tension forces in the supports

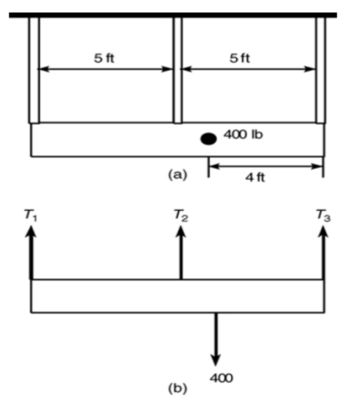
T1 + T2 + T3= 400
10T1+ 5T2 =1600

$$\begin{bmatrix} 1 & 1 & 1 \\ 10 & 5 & 0 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} = \begin{bmatrix} 400 \\ 1600 \end{bmatrix}$$

© McGraw Hill LLC

A Light Fixture And Its Free-body Diagram

```
>> A = [1,1,1;10,5,0];
b = [400;1600];
rank(A)
rank([A, b])
T1 = A\b
T2 = pinv(A)*b

ans =    2
ans =    2

T1 =   160.0000         0      240.0000

T2 =   93.3333      133.3333     173.3333
```



© McGraw Hill LLC

Recall that if $|A| = 0$, the equation set is singular.

If you try to solve a singular set using MATLAB, it prints a message warning that the matrix is singular and does not try to solve the problem.

Use of the `rref` Function

In cases that have an infinite number of solutions, some of the unknowns can be expressed in terms of the remaining unknowns, whose values are arbitrary. We can use the `rref` command to find these relations.

© McGraw Hill LLC

The rref Function

- We can always express some of the unknowns in an underdetermined set as functions of the remaining unknowns. We can obtain such a form by multiplying the set's equations by suitable factors and adding the resulting equations to eliminate an unknown variable.
- The MATLAB rref function provides a procedure to reduce an equation set to this form, which is called the reduced row echelon form.
- The syntax is rref([A b]). The output is the augmented matrix [C d] that corresponds to the equation set Cx = d. This set is in reduced row echelon form.

© McGraw Hill LLC

The rref Function Example

$$x1 + x2 + 5x3 = 12$$
$$4x1 + 2x2 + 8x3 = 28$$
$$3x1 + 2x2 + 9x3 = 26$$

```
>> A=[1 1 5; 4 2 8 ; 3 2 9];
b=[12 28 26]';
rank(A)
rank([A b])
rref([A,b])

ans =    2
ans =    2
ans =
   1    0   -1    2
   0    1    6   10
   0    0    0    0
```

$$x1 - x3 = 2 \qquad x1 = x3 + 2$$
$$x2 + 6x3 = 10 \longrightarrow x2 = -6x3 + 10$$

© McGraw Hill LLC

An *ill-conditioned set* of equations

An ill-conditioned set of equations is a set of equations close to being singular.

The ill-conditioned status depends on the accuracy with which the solution calculations are made.

When the internal numerical accuracy used by MATLAB is insufficient to obtain a solution, MATLAB prints a message to warn you that the matrix is close to singular and that the results might be inaccurate.

© McGraw Hill LLC

An *ill-conditioned set* of equations

$$3x_1 - 4x_2 = 5$$
$$6x_1 - 8x_2 = 2$$

```
>> A = [3,-4;6,-8];
b = [2;5];
x = inv(A)*b
Warning: Matrix is singular to working
precision.
x =     Inf      Inf
```

$$3\,x_1 - 4x_2 = 5$$
$$6.000000000000001x_1 - 8x_2 = 2$$

```
A = [3,-4;6.000000000000001,-8];
b = [2;5];
x = inv(A)*b
Warning: Matrix is close to
singular or badly scaled. Results
may be inaccurate.

RCOND =  1.586033e-17.
x = 1.0e+15 * (1.5012  1.1259)
```

© McGraw Hill LLC

The `pinv` command can obtain a solution of an underdetermined set.

To solve the equation set **Ax** = **b** using the `pinv` command, type

```
x = pinv(A)*b
```

Underdetermined sets have an infinite number of solutions, and the `pinv` command produces a solution that gives the minimum value of the *Euclidean norm,* which is the magnitude of the solution vector **x**.

© McGraw Hill LLC

Overdetermined Systems

An *overdetermined system* is a set of equations that has more independent equations than unknowns.

For such a system the matrix inverse method will not work because the **A** matrix is not square.

However, some overdetermined systems have exact solutions, and they can be obtained with the left division method `x = A\b`.

© McGraw Hill LLC

Overdetermined Systems with an Exact Solution

The left-division method sometimes gives an answer for overdetermined systems, but it does not indicate whether the answer is the exact solution.

We need to check the ranks of A and [A b] to know whether the answer is the exact solution.

To interpret MATLAB answers correctly for an overdetermined system, first check the ranks of A and [A b] to see whether an exact solution exists; if one does not exist, then you know that the left-division answer is a least squares solution.

© McGraw Hill LLC

Overdetermined Systems with an Exact Solution

$$y = -2x + 1$$
$$y = 2x + 1$$
$$y = x + 1$$

$$\begin{bmatrix} -2 & 1 \\ 2 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

```
>> A=[-2 1 ; 2 1; -1 1] ;
b=[1 1 1]' ;
rank(A)
rank([A,b])
ans = 2
ans = 2

pinv(A)*b    %  or A\b
ans =   0.0000
        1.0000
```

© McGraw Hill LLC

Overdetermined systems, With no exact solution exists.

In some of these cases, the left-division method does not yield an answer, while in other cases the left-division method gives an answer that satisfies the equation set only in a “least squares” sense.

When MATLAB gives an answer to an overdetermined set, it does not tell us whether the answer is the exact solution.

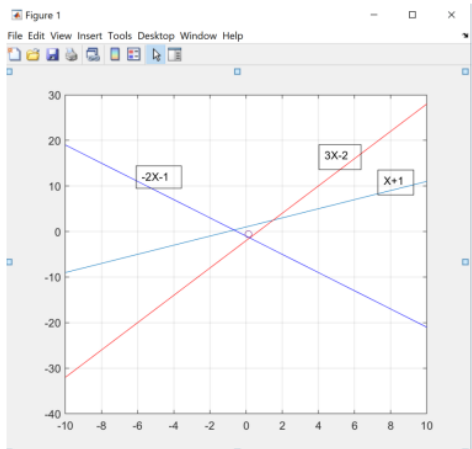
© McGraw Hill LLC

Overdetermined Systems with no Exact Solution

$$y = -2x - 1;$$
$$y = 3x - 2;$$
$$y = x + 1;$$

$$\begin{bmatrix} -2 & -1 \\ 3 & -2 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} -1 \\ -2 \\ 1 \end{bmatrix}$$

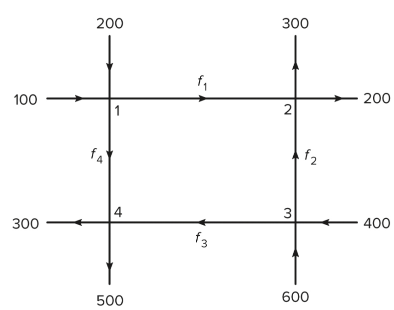
```
A=[2 1 ; -3 1; -1
1] ;
b=[-1 -2 1]' ;
rank(A)
rank([A,b])

ans = 2
ans = 3

pinv(A)*b ;
 % or A\b
ans = 0.1316
      -0.5789
```



© McGraw Hill LLC

Example of an Underdetermined System: A Network of One-Way Streets

A traffic engineer wants to know if measurements of traffic entering and leaving a road network are sufficient to predict the traffic on each street in the network. For example, consider the network of one-way streets here. The numbers shown are the measured traffic in vehicles per hour . Assume that no vehicles park anywhere within the network. If possible, calculate the traffic $f1$, $f2$, $f3$, and $f4$. If this is not possible, suggest how to obtain the necessary information.



© McGraw Hill LLC

Example: A Network of One-way Streets

The flow into intersection 1 must equal the flow out of the intersection.

This gives $100 + 200 = f1 + f4$

Similarly, for the other three intersections, we have

$f1 + f2 = 300 + 200$

$600 + 400 = f2 + f3$

$f3 + f4 = 300 + 500$

© McGraw Hill LLC

Example: A Network of One-way Streets

$$\begin{aligned} f1 + f4 &= 100 + 200 \\ f1 + f2 &= 300 + 200 \\ f2 + f3 &= 600 + 400 \\ f3 + f4 &= 300 + 500 \end{aligned}$$

Putting these in the matrix form **Ax= b**

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \qquad \mathbf{b} = \begin{bmatrix} 300 \\ 500 \\ 1000 \\ 800 \end{bmatrix} \qquad \mathbf{x} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}$$

First, check the ranks of A and [A b], using the MATLAB rank function.

rank(A)= rank([A, b]) $\longrightarrow$ 3

The number of unknowns > rank (A)

Underdetermined system with **infinite number of solutions**

© McGraw Hill LLC

Example: A Network of One-way Streets

```
> A=[1 0 0 1 ; 1 1 0 0 ; 0 1 1 0 ; 0 0 1 1] ;
> b =[300 ;500; 1000; 800]
```

$$\boldsymbol{Ax} = \boldsymbol{b}$$

The `rref([A b])` function produces the reduced augmented matrix

```
>> rref ([A b])
ans =
    1    0    0    1   300
    0    1    0   -1   200
    0    0    1    1   800
    0    0    0    0     0
```

which corresponds to the reduced system

$$f1 + f4 = 300$$
$$f2 - f4 = 200$$
$$f3 + f4 = 800$$

© McGraw Hill LLC

Example: A Network of One-way Streets

$$f1 + f4 = 300$$
$$f2 - f4 = 200$$
$$f3 + f4 = 800$$

These can be solved easily as follows:

$$f1 = 300 - f4$$
$$f2 = 200 + f4$$
$$f3 = 800 - f4$$

If we could measure the flow on one of the internal roads, say f4, then we could compute the other flows.

We recommend that the engineer arrange to have this additional measurement made.

© McGraw Hill LLC

Solving Linear Equations: Summary

If the number of equations in the set *equals* the number of unknown variables, the matrix **A** is square, and MATLAB provides two ways of solving the equation set **Ax** = **b**:

1. The matrix inverse method; solve for **x** by typing `x = inv(A)*b`.
2. The matrix left-division method; solve for **x** by typing `x = A\b`.

© McGraw Hill LLC

Solving Linear Equations: Summary

If **A** is square and if MATLAB does not generate an error message when you use one of these methods, then the set has a unique solution, which is given by the left-division method.

You can always check the solution for `x` by typing `A*x` to see if the result is the same as `b`.

© McGraw Hill LLC

Solving Linear Equations: Summary

If you receive an error message, the set is underdetermined, and either it does not have a solution or it has more than one solution.

In such a case, if you need more information, you must use the following procedures.

© McGraw Hill LLC

Solving Linear Equations: Summary

For underdetermined and over-determined sets, MATLAB provides three ways of dealing with the equation set **Ax** = **b.** (Note that the matrix inverse method will never work with such sets.)

1. The matrix left-division method; solve for **x** by typing `x = A\b`.
2. The pseudo-inverse method; solve for **x** by typing `x = pinv(A)*b`.
3. The reduced row echelon form (RREF) method. This method uses the MATLAB function `rref` to obtain a solution.

© McGraw Hill LLC

Solving Linear Equations: Summary

Underdetermined Systems

In an *underdetermined* system not enough information is given to determine the values of all the unknown variables.

- An infinite number of solutions might exist in which one or more of the unknowns are dependent on the remaining unknowns.
- For such systems the matrix inverse method will not work because either **A** is not square or because $|\mathbf{A}| = 0$.

© McGraw Hill LLC

Solving Linear Equations: Summary

The left-division method will give a solution with some of the unknowns arbitrarily set equal to zero, but this solution is not the general solution.

An infinite number of solutions might exist even when the number of equations equals the number of unknowns. The left-division method fails to give a solution in such cases.

In cases that have an infinite number of solutions, some of the unknowns can be expressed in terms of the remaining unknowns, whose values are arbitrary. The `rref` function can be used to find these relations.

© McGraw Hill LLC

Solving Linear Equations: Summary

Overdetermined Systems

An *overdetermined* system is a set of equations that has more independent equations than unknowns.

- For such a system Cramer's method and the matrix inverse method will not work because the **A** matrix is not square.
- Some overdetermined systems have exact solutions, which can be obtained with the left-division method `A\b`.

© McGraw Hill LLC

Solving Linear Equations: Summary

For overdetermined systems that have no exact solution, the answer given by the left-division method satisfies the equation set only in a least squares sense.

When we use MATLAB to solve an overdetermined set, the program does not tell us whether the solution is exact. We must determine this information ourselves. The first step is to check the ranks of **A** and [**A b**] to see whether a solution exists; if no solution exists, then we know that the left-division solution is a least squares answer.

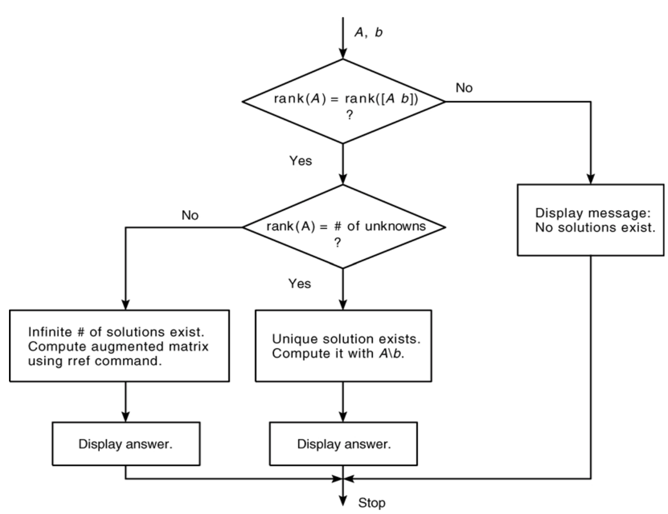
© McGraw Hill LLC

Pseudocode for the Linear Equation Solver

1. If the rank of **A** equals the rank of [**A b**], then determine whether the rank of **A** equals the number of unknowns. If so, there is a unique solution, which can be computed using left division. Display the results and stop.
2. Otherwise, there is an infinite number of solutions, which can be found from the augmented matrix. Display the results and stop.
3. Otherwise (if the rank of **A** does not equal the rank of [**A b**]), then there are no solutions. Display this message and stop.

© McGraw Hill LLC

Flowchart of the Linear Equation Solver



© McGraw Hill LLC

MATLAB Program to Solve Linear Equations

```
% Script file lineq.m
% Solves the set Ax = b, given A and b.
% Check the ranks of A and [A b].
if rank(A) == rank([A b])
   % The ranks are equal.
   size_A = size(A);
   % Does the rank of A equal the number of unknowns?
   if rank(A) == size_A(2)
    % Yes. Rank of A equals the number of unknowns.
    disp('There is a unique solution, which is:')
    x = A\b % Solve using left division.
   else
    % Rank of A does not equal the number of unknowns.
    disp('There is an infinite number of solutions.')
    disp('The augmented matrix of the reduced system is:')
    rref([A b]) % Compute the augmented matrix.
   end
else
    % The ranks of A and [A b] are not equal.
    disp('There are no solutions.')
end
```

© McGraw Hill LLC

Because learning changes everything.®

www.mheducation.com

© McGraw Hill LLC. All rights reserved. No reproduction or distribution without the prior written consent of McGraw Hill LLC.

Lesson Explainer: Rank of a Matrix: Determinants

https://www.nagwa.com/en/explainers/402106373582/#:~:text=The%20%E2%80%9Crank%E2%80%9D%20of%20a%20matrix,%F0%9D%91%9A%20or%20%F0%9D%91%9B%20is%20smaller.

© McGraw Hill LLC