

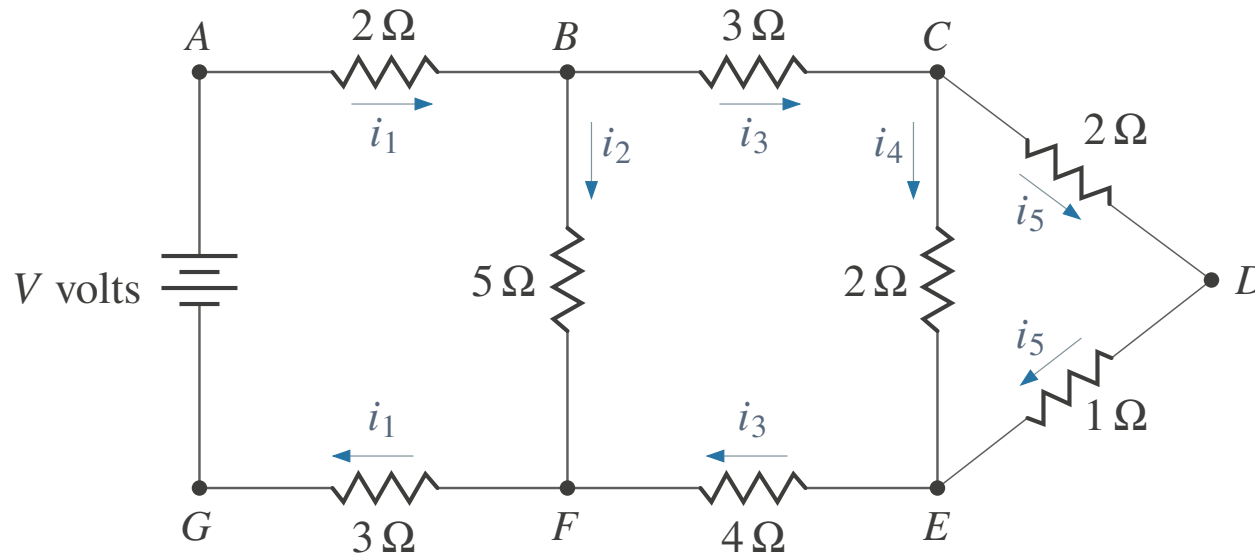
Lecture 4

Linear Systems of Equations, Part 1

Virginia Tech Department of Computer Science
Fall 2026 • Sumeet Khatri

Electrical circuit equations

Using Kirchoff's laws, we obtain the following equations



$$5i_1 + 5i_2 + 0i_3 + 0i_4 + 0i_5 = V,$$

$$0i_1 + 0i_2 + i_3 - i_4 - i_5 = 0,$$

$$0i_1 + 0i_2 + 0i_3 + 2i_4 - 3i_5 = 0,$$

$$i_1 - i_2 - i_3 + 0i_4 + 0i_5 = 0,$$

$$0i_1 + 5i_2 - 7i_3 - 2i_4 + 0i_5 = 0$$

Electrical circuit equations as a linear system

$$\begin{pmatrix} 5 & 5 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 \\ 0 & 0 & 0 & 2 & -3 \\ 1 & -1 & -1 & 0 & 0 \\ 0 & 5 & -7 & -2 & 0 \end{pmatrix} \begin{pmatrix} i_1 \\ i_2 \\ i_3 \\ i_4 \\ i_5 \end{pmatrix} = \begin{pmatrix} V \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

$$A\mathbf{i} = \mathbf{b}, \quad A \in \mathbb{R}^{5 \times 5}$$

Rows record equations. Columns record unknown currents.

The coefficients and applied voltage are given; the five currents are to be found.

Matrices and vectors

$$\mathbf{x} \in \mathbb{R}^n, \quad A \in \mathbb{R}^{m \times n}, \quad a_{ij} = \text{entry in row } i, \text{ column } j$$

Addition acts entry by entry and requires equal shapes. Scalar multiplication scales every entry.

Coordinate vectors and the identity

$$\mathbf{e}_j = \begin{pmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix}, \quad I_n = (\mathbf{e}_1 \cdots \mathbf{e}_n) \quad I_3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Two ways to read a matrix–vector product

$$A \in \mathbb{R}^{m \times n}, \quad \mathbf{x} \in \mathbb{R}^n \quad \implies \quad A\mathbf{x} \in \mathbb{R}^m$$

One output from each row

$$(A\mathbf{x})_i = \sum_{j=1}^n a_{ij}x_j$$

A weighted combination of columns

$$A\mathbf{x} = x_1\mathbf{a}_1 + \cdots + x_n\mathbf{a}_n$$

Solving the system asks which column weights produce the requested right-hand side.

$$A(\alpha\mathbf{x} + \beta\mathbf{y}) = \alpha A\mathbf{x} + \beta A\mathbf{y}$$

Matrix multiplication

$$A_{m \times n} B_{n \times p} = (AB)_{m \times p}, \quad (AB)_{ij} = \sum_{k=1}^n a_{ik} b_{kj}$$

$$B = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \quad C = \begin{pmatrix} 1 & 0 \\ 1 & 1 \end{pmatrix}$$

$$BC = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix} \neq \begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} = CB$$

Associativity still holds: $(AB)x = A(Bx)$.

Transpose reverses rows, columns, and product order

$$A \in \mathbb{R}^{m \times n} \implies A^T \in \mathbb{R}^{n \times m}, \quad (A^T)_{ji} = a_{ij}$$

$$(A^T)^T = A, \quad (A + B)^T = A^T + B^T$$

$$(AB)^T = B^T A^T$$

Solving a linear system of equations

$$Ax = b, \quad A \in \mathbb{R}^{m \times n}$$

Existence and uniqueness

Do the equations have a common solution? If so, is it the only one?

Accuracy

Can a particular procedure fail or lose significant digits?

Computation

How much arithmetic and storage are needed? How does the cost grow with the system size?

Example: two equations, two unknowns

$$\begin{aligned}2x + y &= 5, \\ 4x + 3y &= 11.\end{aligned}$$

Solve the second equation for one variable in terms of the other.

$$x = \frac{11 - 3y}{4}$$

Substitute this expression into the first equation to eliminate x .

Eliminate one variable, then solve for the other

$$2\left(\frac{11 - 3y}{4}\right) + y = 5$$

$$11 - 3y + 2y = 10 \quad \implies \quad y = 1, \quad x = \frac{11 - 3}{4} = 2$$

Gaussian elimination formalizes this idea for larger systems: reduce the unknowns, then substitute back.

Gaussian elimination: row operations to determine the solution

Three reversible operations

Exchange rows: $R_i \leftrightarrow R_j$.

Scale a row: $R_i \leftarrow cR_i$, with $c \neq 0$.

Add a multiple: $R_i \leftarrow R_i + cR_j$, with $i \neq j$.

Use the entire augmented row

$$(A \mid \mathbf{b}) \rightarrow (\tilde{A} \mid \tilde{\mathbf{b}})$$

The same unknown vectors solve the original and transformed systems.

$$\{\mathbf{x} : A\mathbf{x} = \mathbf{b}\} = \{\mathbf{x} : \tilde{A}\mathbf{x} = \tilde{\mathbf{b}}\}$$

Row operations for a 3x3 system

$$\left(\begin{array}{ccc|c} 2 & 1 & -1 & 8 \\ -3 & -1 & 2 & -11 \\ -2 & 1 & 2 & -3 \end{array} \right)$$

$$m_{21} = \frac{-3}{2}, \quad m_{31} = \frac{-2}{2} = -1$$

$$R_2 \leftarrow R_2 - \left(-\frac{3}{2} \right) R_1, \quad R_3 \leftarrow R_3 - (-1)R_1$$

The **pivot element** is 2. These multipliers eliminate the entries below it.

Row operations for a 3x3 system

$$\left(\begin{array}{ccc|c} 2 & 1 & -1 & 8 \\ 0 & 1/2 & 1/2 & 1 \\ 0 & 2 & 1 & 5 \end{array} \right)$$

$$\text{New second RHS: } -11 - \left(-\frac{3}{2}\right)8 = 1$$

$$m_{32} = \frac{2}{1/2} = 4, \quad R_3 \leftarrow R_3 - 4R_2$$

The next pivot comes from the updated matrix.

Triangular form reveals the solution from the bottom

$$\left(\begin{array}{ccc|c} 2 & 1 & -1 & 8 \\ 0 & 1/2 & 1/2 & 1 \\ 0 & 0 & -1 & 1 \end{array} \right)$$

$$x_3 = -1, \quad \frac{1}{2}x_2 + \frac{1}{2}(-1) = 1 \implies x_2 = 3$$

$$2x_1 + 3 - (-1) = 8 \implies \mathbf{x} = \begin{pmatrix} 2 \\ 3 \\ -1 \end{pmatrix}$$

Forward elimination makes the system triangular; **back substitution** recovers the unknowns.

One update rule performs every forward step

$$k = 1, \dots, n - 1, \quad i = k + 1, \dots, n$$

$$m_{ik} = \frac{a_{ik}}{a_{kk}}$$

$$\begin{aligned} a_{ij} &\leftarrow a_{ij} - m_{ik}a_{kj}, & j = k, \dots, n, \\ b_i &\leftarrow b_i - m_{ik}b_k. \end{aligned}$$

Save the multiplier before overwriting its coefficient. Every divisor must be nonzero.

Back substitution repeats one equation upward

$$x_n = \frac{b_n}{a_{nn}}$$

$$x_i = \frac{b_i - \sum_{j=i+1}^n a_{ij}x_j}{a_{ii}}, \quad i = n, n-1, \dots, 1$$

When $i = n$, the sum is empty and equals zero.

At each higher row, the unknowns to its right have already been recovered.

(In)consistent linear systems

A solution is one vector whose entries satisfy every equation simultaneously.

$$\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^n : A\mathbf{x} = \mathbf{b}\}$$

Consistent

At least one solution exists: $\mathcal{S} \neq \emptyset$

$$x_1 + x_2 = 1: \quad (x_1, x_2) = (1 - t, t)$$

Inconsistent

No common solution exists: $\mathcal{S} = \emptyset$

$$\begin{aligned} x_1 + x_2 &= 1, \\ x_1 + x_2 &= 2. \end{aligned}$$

Consistency asks about existence. A consistent system may have one solution or infinitely many.

Dependent equations can agree

$$\left(\begin{array}{cc|c} 1 & 1 & 3 \\ 2 & 2 & c \end{array} \right) \xrightarrow{R_2 \leftarrow R_2 - 2R_1} \left(\begin{array}{cc|c} 1 & 1 & 3 \\ 0 & 0 & c - 6 \end{array} \right)$$

If $c = 6$, the second equation becomes $0 = 0$. Only one independent restriction remains.

$$x_2 = t, \quad x_1 = 3 - t, \quad t \in \mathbb{R}$$

$$\mathbf{x} = \begin{pmatrix} 3 \\ 0 \end{pmatrix} + t \begin{pmatrix} -1 \\ 1 \end{pmatrix}$$

The same coefficients can contradict each other

$$c = 7: \quad \left(\begin{array}{cc|c} 1 & 1 & 3 \\ 0 & 0 & 1 \end{array} \right)$$

$$0 = 1 \quad \implies \quad \text{no solution}$$

The original equations now describe distinct parallel lines.

A rectangular system

$$\left(\begin{array}{ccc|c} 1 & 2 & -1 & 3 \\ 2 & 4 & 1 & 9 \end{array} \right) \xrightarrow{R_2 \leftarrow R_2 - 2R_1} \left(\begin{array}{ccc|c} 1 & 2 & -1 & 3 \\ 0 & 0 & 3 & 3 \end{array} \right)$$

Skip coefficient column 2. Keep row 2 and use its entry in column 3 as the next pivot.

Pivot variables: x_1, x_3 Free variable: x_2

A non-pivot column need not be an all-zero column.

Choose the free variable, then substitute back

$$3x_3 = 3 \implies x_3 = 1$$

$$x_2 = t, \quad x_1 + 2t - 1 = 3 \implies x_1 = 4 - 2t$$

$$\mathbf{x} = \begin{pmatrix} 4 \\ 0 \\ 1 \end{pmatrix} + t \begin{pmatrix} -2 \\ 1 \\ 0 \end{pmatrix}, \quad t \in \mathbb{R}$$

Another rectangular system

$$\begin{aligned}x_1 + x_2 &= 3, \\ 2x_1 - x_2 &= 0, \\ 3x_1 &= c.\end{aligned}$$

$$\left(\begin{array}{cc|c} 1 & 1 & 3 \\ 0 & -3 & -6 \\ 0 & -3 & c-9 \end{array} \right) \xrightarrow{R_3 \leftarrow R_3 - R_2} \left(\begin{array}{cc|c} 1 & 1 & 3 \\ 0 & -3 & -6 \\ 0 & 0 & c-3 \end{array} \right)$$

Both coefficient columns have pivots. The last row still has to be satisfied.

A pivot in every column means at most one solution

$$\left(\begin{array}{cc|c} 1 & 1 & 3 \\ 0 & -3 & -6 \\ 0 & -3 & c-9 \end{array} \right) \xrightarrow{R_3 \leftarrow R_3 - R_2} \left(\begin{array}{cc|c} 1 & 1 & 3 \\ 0 & -3 & -6 \\ 0 & 0 & c-3 \end{array} \right)$$

Compatible right-hand side

$$c = 3: \quad 0 = 0$$

$$x_2 = 2, \quad x_1 = 1$$

Incompatible right-hand side

$$c \neq 3: \quad 0 = c - 3$$

There is no exact solution.

An inconsistent rectangular system

$$\left(\begin{array}{ccc|c} 1 & -1 & 2 & 2 \\ 3 & -3 & 6 & c \end{array} \right) \rightarrow \left(\begin{array}{ccc|c} 1 & -1 & 2 & 2 \\ 0 & 0 & 0 & c-6 \end{array} \right)$$

If $c = 7$

$$0 = 1$$

No choices of the three unknowns work.

If $c = 6$

$$x_2 = s, \quad x_3 = t, \quad x_1 = 2 + s - 2t$$

Two free parameters describe a plane.

A rectangular system with infinitely many solutions

$$\left(\begin{array}{cc|c} 1 & 2 & 3 \\ 2 & 4 & 6 \\ 3 & 6 & 9 \end{array} \right) \rightarrow \left(\begin{array}{cc|c} 1 & 2 & 3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{array} \right)$$

$$x_1 + 2x_2 = 3 \quad \implies \quad \mathbf{x} = \begin{pmatrix} 3 - 2t \\ t \end{pmatrix}, \quad t \in \mathbb{R}$$

Gaussian elimination ordering may need to be changed!

$$0x_1 + x_2 = 1$$

$$x_1 + x_2 = 2$$

The first candidate pivot is 0. Dividing by it is impossible.

$$R_1 \leftrightarrow R_2 \quad \Longrightarrow \quad \begin{array}{l} x_1 + x_2 = 2 \\ x_2 = 1 \end{array}$$

$$x_2 = 1, \quad x_1 = 1$$

A zero candidate pivot does not, by itself, mean that the system cannot be solved.

Pitfalls with floating-point numbers

$$\begin{aligned} \varepsilon x_1 + x_2 &= 1 \\ x_1 + x_2 &= 2 \end{aligned} \quad 0 < |\varepsilon| \ll 1$$

Without reordering, the multiplier is $1 / \varepsilon$. Then, doing $R_2 \leftarrow R_2 - \varepsilon^{-1} R_1$, we obtain

$$(1 - \varepsilon^{-1})x_2 = 2 - \varepsilon^{-1} \Rightarrow x_2 = \frac{2 - \varepsilon^{-1}}{1 - \varepsilon^{-1}}$$

Finite-precision failure

$$\hat{x}_2 \approx 1, \quad \hat{x}_1 = \varepsilon^{-1}(1 - \hat{x}_2) \approx 0$$

Exact solution

$$x_1 = \frac{1}{1 - \varepsilon}, \quad x_2 = \frac{1 - 2\varepsilon}{1 - \varepsilon}$$

The exact solution approaches two ones. On the other hand, the computed first component can be zero.

Pitfalls with floating-point numbers

$$\varepsilon = 10^{-9}, \quad \varepsilon^{-1} = 10^9$$

An eight-significant-digit decimal machine may use a longer accumulator, then round when storing the result.

$$10^9 - 2 = 0.099999999980000000 \times 10^{10}$$

$$fl_8(10^9 - 2) = 0.10000000 \times 10^{10} = 10^9$$

The same rounding occurs for $10^9 - 1$. Their rounded ratio is 1, so back substitution returns $\hat{x}_1 = 0$.

Use the equation with a stronger first coefficient

$$\begin{aligned}x_1 + x_2 &= 2 \\ \varepsilon x_1 + x_2 &= 1\end{aligned}$$

$$R_2 \leftarrow R_2 - \varepsilon R_1 \quad \Longrightarrow \quad (1 - \varepsilon)x_2 = 1 - 2\varepsilon$$

$$x_2 = \frac{1 - 2\varepsilon}{1 - \varepsilon} \approx 1, \quad x_1 = 2 - x_2 \approx 1$$

The pivot equation now recovers the first component without subtracting two nearly equal numbers.

Scaling and changing row order

$$s_i = \max_{1 \leq j \leq n} |a_{ij}|, \quad \ell = (1, 2, \dots, n)$$

At stage k , select the first largest candidate ratio:

$$\left\{ \frac{|a_{\ell_i, k}|}{s_{\ell_i}} : i = k, \dots, n \right\}$$

Exchange the chosen index with ℓ_k , then use the multiplier

$$m_{\ell_i, k} = \frac{a_{\ell_i, k}}{a_{\ell_k, k}}, \quad i = k + 1, \dots, n$$

The scale vector stays fixed. Only entries of ℓ are exchanged.

Cost of linear algebra operations

One scalar addition, subtraction, multiplication, or division counts as one floating-point operation: one *flop*.

Operation	Flops
Length- n vector addition	n
$\mathbf{x}^\top \mathbf{y}$	$2n - 1$
$\mathbf{y} \leftarrow \alpha \mathbf{x} + \mathbf{y}$	$2n$
$\mathbf{u}\mathbf{v}^\top$ (m by n)	mn

Describing scaling with asymptotic notation

Let $f(n) \geq 0$ and $g(n) > 0$ eventually. For all sufficiently large n , choose constants $c, C > 0$ independent of n :

Notation	Required inequality	Meaning
$f(n) = O(g(n))$	$f(n) \leq Cg(n)$	Upper bound
$f(n) = \Omega(g(n))$	$cg(n) \leq f(n)$	Lower bound
$f(n) = \Theta(g(n))$	$cg(n) \leq f(n) \leq Cg(n)$	Tight bound

$$n^2 = O(n^3), \quad n^2 \neq \Theta(n^3)$$

$$T(n) = 2n^3 - n^2, \quad n^3 \leq T(n) \leq 2n^3 \Rightarrow T(n) = \Theta(n^3)$$

Cost of matrix operations

Matrix times vector

$$A \in \mathbb{R}^{m \times n}, \quad \mathbf{x} \in \mathbb{R}^n$$

$$T_{A\mathbf{x}} = m(2n - 1) = 2mn - m$$

Compute m dot products of length n .

Matrix times matrix

$$A \in \mathbb{R}^{m \times k}, \quad B \in \mathbb{R}^{k \times n}$$

$$T_{AB} = mn(2k - 1)$$

Compute mn dot products of length k .

$$T_{AB} = 2n^3 - n^2 \sim 2n^3 \quad \text{for two } n \times n \text{ matrices}$$

Associativity can save orders of magnitude of work

$$A: 1000 \times 10, \quad B: 10 \times 1000, \quad \mathbf{x}: 1000 \times 1$$

Form the matrix product first

$$(AB)\mathbf{x}$$

$$19,000,000 + 1,999,000$$

$$= 20,999,000 \text{ flops}$$

One million intermediate entries.

Form the vector product first

$$A(B\mathbf{x})$$

$$19,990 + 19,000$$

$$= 38,990 \text{ flops}$$

Ten intermediate entries.

The exact results agree. Floating-point results can differ slightly because rounding does not preserve associativity.

Strassen's algorithm

There is an algorithm that does better than $\Theta(n^3)$!

$n \times n$ matrices with even n : $A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \quad B = \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}$

Partition both matrices into four square blocks of size $(n/2) \times (n/2)$.

$$M_1 = (A_{11} + A_{22})(B_{11} + B_{22})$$

$$M_2 = (A_{21} + A_{22})B_{11}$$

$$M_3 = A_{11}(B_{12} - B_{22})$$

$$M_4 = A_{22}(B_{21} - B_{11})$$

$$M_5 = (A_{11} + A_{12})B_{22}$$

$$M_6 = (A_{21} - A_{11})(B_{11} + B_{12})$$

$$M_7 = (A_{12} - A_{22})(B_{21} + B_{22})$$

The order of factors is preserved, so these formulas work for matrix blocks.

The tradeoff is fewer block multiplications and more block additions.

Strassen's algorithm

$$C = AB = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} \quad \begin{aligned} C_{11} &= M_1 + M_4 - M_5 + M_7, & C_{12} &= M_3 + M_5 \\ C_{21} &= M_2 + M_4, & C_{22} &= M_1 - M_2 + M_3 + M_6 \end{aligned}$$

$$T(n) = 7T(n/2) + 18(n/2)^2, \quad T(1) = 1$$

$$T(n) = 7n^{\log_2 7} - 6n^2 = \Theta(n^{\log_2 7})$$

$$\log_2 7 \approx 2.8074 < 3$$

Improving the exponent

$$\omega = \inf \left\{ \tau \geq 0 : \text{an algorithm uses } O(n^\tau) \text{ arithmetic steps} \right\}$$

Algorithm / result	Exponent	Notes
Classical	3	Ordinary arithmetic
Strassen (1969)	$\log_2 7 \approx 2.8074$	7 products per 2×2 block
AlphaTensor (2022)	$\log_4 47 \approx 2.7773$	Arithmetic modulo 2
AlphaEvolve (2025)	$\log_4 48 \approx 2.7925$	48-product complex formula
Dumas–Pernet–Sedoglavic (2025)	$\log_4 48 \approx 2.7925$	48-product formula applicable to real arithmetic
Dumas et al. (2026)	$\log_4 48 \approx 2.7925$	Improved leading constant; same exponent
Alman et al. (2025)	< 2.371339	Theoretical asymptotic bound
Dupont et al., with AlphaEvolve (2026)	< 2.371177	Improved theoretical bound

Other recent developments



2026-08-18

Improving the matrix multiplication exponent with modern optimization and AlphaEvolve

Emilien Dupont^{*,1}, Marvin Eisenberger^{*,1}, Borislav Kozlovskii^{*,1}, Abbas Mehrabian^{*,1}, Francisco J. R. Ruiz^{*,1}, Abigail See^{*,1}, Renfei Zhou^{*,2}, Josh Alman³, Virginia Vassilevska Williams⁴ and Matej Balog¹

^{*}Equal contribution in alphabetical order, ¹Google DeepMind, ²Carnegie Mellon University, ³Columbia University, ⁴MIT

<https://arxiv.org/abs/2608.16884>

August 2026!