📘

Parametric Representations of Lines

Source: Parametric representations of lines (video) | Khan Academy

💡 Introduction: From Slopes to Vectors

You might be wondering—why go through all this “linear algebra” when you already know how to draw lines from Algebra 1? You’ve seen:

$$y = mx + b$$

Simple, right? So why all the vectors and sets? 😩

Because traditional algebra is built around **2D thinking**. But in linear algebra, we move beyond! We need a method that works in:

- $\mathbb{R}^2$: Flat, cozy, familiar 📊
- $\mathbb{R}^3$: Welcome to 3D—spacey 🚀
- $\mathbb{R}^n$: Up to 50, 100, or infinite dimensions 🧠🌌

To represent a line *anywhere*, *in any dimension*, you need the **parametric form of a line**, powered by **vectors** and **scalar multiplication**. Let's dive in! 🌊

🧷 Lines Through the Origin in $\mathbb{R}^2$

Defining a Vector

Let's start with a simple 2D vector:

$$\vec{v} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}$$

In standard position (meaning: tail at the origin $(0, 0)$), it points:

- 📍 2 units right (x-direction)
- 📍 1 unit up (y-direction)

So visually, it points diagonally, rising 1 for every 2 steps across—a slope of $\frac{1}{2}$.

Set of Scalar Multiples

Now let's define a **set**:

$$S = \{c \cdot \vec{v} \mid c \in \mathbb{R}\}$$

This means: we're multiplying $\vec{v}$ by every real number c, stretching or flipping it depending on c's value.

Examples 🎨

- $c = 2$:

  $$2 \cdot \vec{v} = \begin{bmatrix} 4 \\ 2 \end{bmatrix}$$

  A longer vector in the same direction.

- $c = 1.5$:

  $$\begin{bmatrix} 3 \\ 1.5 \end{bmatrix}$$

  Slightly shorter—still collinear.

- $c = 0.001$:

  Tiny vector pointing the same way.

- $c = -10$:

Very long vector, but reversed 🚫➡️

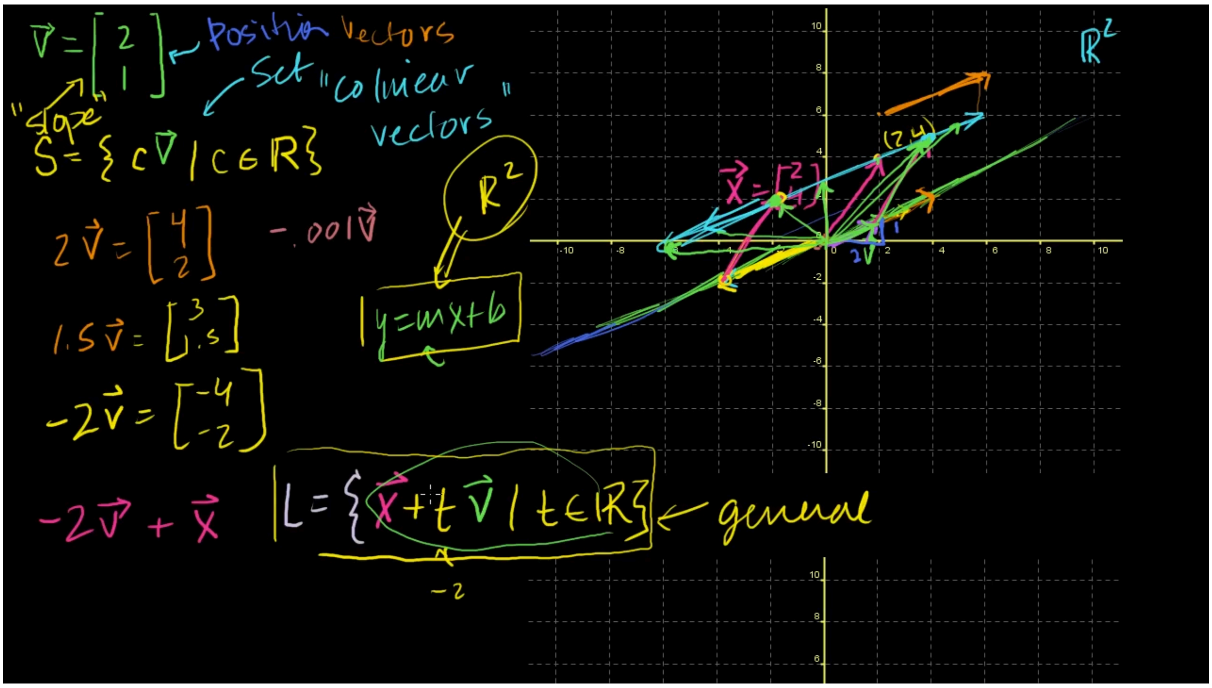
Geometric Insight 🧭

All such vectors are **collinear**—they lie along the same line through the origin. If we view them as **position vectors** (i.e., they point *to* a coordinate), then together they **trace out an infinite line**:

$$\text{Line of slope } \frac{1}{2} \text{ passing through the origin}$$

💡 Drawing them in **standard position** (tail at origin) is crucial to see this collinearity clearly.

🔀 Lines Not Through the Origin



Let’s level up: what if we want the **same line**, but it **doesn’t** pass through the origin?

Shifting the Line

Let's pick a new **starting point**:

$$\vec{x} = \begin{bmatrix} 2 \\ 4 \end{bmatrix}$$

This vector points to a position elsewhere on the plane. We now want a line **parallel** to our original, but passing through $\vec{x}$.

Parametric Equation of the Line

We define a new set:

$$L = \{\vec{x} + t \cdot \vec{v} \mid t \in \mathbb{R}\}$$

- $\vec{v}$: direction vector (defines slope)
- $\vec{x}$: starting point (shifts the line)
- t: parameter (scalar, "time" slider ⏱️)

Visual Breakdown 🧠✨

- $t = 0 \Rightarrow$ You're exactly at $\vec{x}$
- $t = 1 \Rightarrow$ You move in direction $\vec{v}$
- $t = -1 \Rightarrow$ You move opposite to $\vec{v}$
- Varying t fills the entire line through $\vec{x}$, parallel to $\vec{v}$

🎯 Think of $\vec{v}$ as a "directional arrow" and $\vec{x}$ as your **starting location**. Together, they sweep out a line in $\mathbb{R}^2$.

But Wait... Why Not Just Use $y = mx + b$?

Because that only works when:

- You're in 2D
- The line can be expressed explicitly as y in terms of x

It **doesn't generalize**.

But our vector form?

$L = \{\vec{x} + t \cdot \vec{v} \mid t \in \mathbb{R}\}$

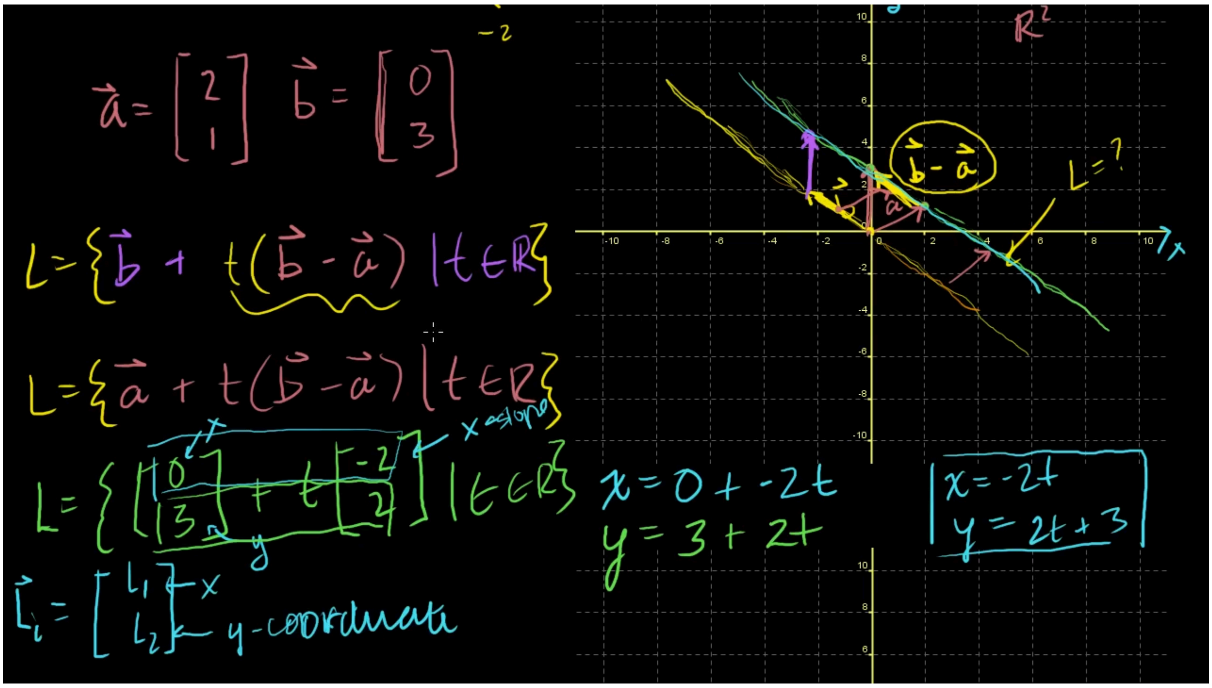
✔️ Works in **any dimension**

✔️ Describes lines **through any point**

✔️ Handles **any direction**

✔️ Becomes essential in $\mathbb{R}^3$ or higher

📍📍 Lines Through Two Points in $\mathbb{R}^2$



Let's say we're given **two position vectors** (a.k.a. points):

$$\vec{a} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \quad \vec{b} = \begin{bmatrix} 0 \\ 3 \end{bmatrix}$$

We want to find a **parametric equation** of the line passing through both points.

Direction Vector = Their Difference

To get the direction the line travels, we subtract:

$\vec{d} = \vec{b} - \vec{a} = \begin{bmatrix} 0 \\ 3 \end{bmatrix} - \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} -2 \\ 2 \end{bmatrix}$ (or even $\vec{a} - \vec{b}$, it will be the same)

This vector tells us the "tilt" or "flow" of the line—its **direction**.

> ⚡ Think of $\vec{d}$ as the bridge that connects $\vec{a}$ and $\vec{b}$.

Parametric Line: Two Equivalent Forms

We can start from either point and walk along $\vec{d}$:

Option 1:

$$L = \left\{ \vec{a} + t \cdot (\vec{b} - \vec{a}) \mid t \in \mathbb{R} \right\}$$

Option 2:

$$L = \left\{ \vec{b} + t \cdot (\vec{b} - \vec{a}) \mid t \in \mathbb{R} \right\}$$

They trace the **same** line, just starting from different launch pads 🚀.

💭 **Personal Insight: Understanding Parametric Lines**

$$L = \left\{ \vec{b} + t \cdot \vec{d} \mid t \in \mathbb{R} \right\}$$

The image above helped me visualize how the parts work together:

- 🔁 $\vec{d}$ **lies *along* the line**. I realized we can compute it as $\vec{b} - \vec{a}$ (or the other way around). It's the "direction vector"—giving us the **slope or flow** of the line.
- 🌀 **The parameter** t lets us **stretch** $\vec{d}$, **reverse** it, or scale it—moving us forward and backward along the line.
- 🧭 Adding $\vec{d}$ to $\vec{b}$ (or $\vec{a}$) lets us **shift the line away from the origin**, anchoring it at a specific place in space.

Split into Coordinates (Parametric Form)

Let’s use Option 2:

$$\vec{b} = \begin{bmatrix} 0 \\ 3 \end{bmatrix}, \quad \vec{d} = \begin{bmatrix} -2 \\ 2 \end{bmatrix}$$

Then:

$$\vec{l}(t) = \vec{b} + t \cdot \vec{d} = \begin{bmatrix} 0 \\ 3 \end{bmatrix} + t \cdot \begin{bmatrix} -2 \\ 2 \end{bmatrix} = \begin{bmatrix} -2t \\ 2t + 3 \end{bmatrix}$$

So we have the coordinate-wise **parametric equations**:

$$x(t) = -2t, \quad y(t) = 2t + 3$$

🎯 This is the **same line** you'd get with slope-intercept form in Algebra 1—but now it's **powered by vectors**, and **ready for higher dimensions**.

🪐 Parametric Lines in $\mathbb{R}^3$

Now for the magic trick: let’s define a line in 3D 🌌.

Let’s say we have two points (position vectors):

$$\vec{P}_1 = \begin{bmatrix} -1 \\ 2 \\ 7 \end{bmatrix}, \quad \vec{P}_2 = \begin{bmatrix} 0 \\ 3 \\ 4 \end{bmatrix}$$

Step 1: Direction Vector

$$\vec{d} = \vec{P}_1 - \vec{P}_2 = \begin{bmatrix} -1 \\ -1 \\ 3 \end{bmatrix}$$

Step 2: Parametric Vector Form

Start at $\vec{P}_1$, and walk in direction $\vec{d}$:

$$L = \left\{ \vec{P}_1 + t \cdot \vec{d} \mid t \in \mathbb{R} \right\}$$

Which is:

$$\vec{l}(t) = \begin{bmatrix} -1 \\ 2 \\ 7 \end{bmatrix} + t \cdot \begin{bmatrix} -1 \\ -1 \\ 3 \end{bmatrix}$$

Step 3: Parametric Equations (Component-wise)

$x(t) = -1 - t$

$y(t) = 2 - t$

$z(t) = 7 + 3t$

🎮 You now have a **precise, controllable formula** to navigate along the line in 3D. This is how a drone, a camera, or a simulated particle travels through space!

> ❗ In $\mathbb{R}^3$, you can't describe a line with one equation like $ax + by + cz = d.$
>
> That would define a **plane**!
>
> You **must** use parametric equations for lines in 3D and beyond.

🚀 Beyond 3D: Higher-Dimensional Lines

Let's jump to $\mathbb{R}^n$, where $n \geq 4$, 10, or even 100 📈.

You can't draw these lines anymore—but mathematically, they're completely valid.

General Parametric Line Form in $\mathbb{R}^n$

Let:

- $\vec{p}$: a point in $\mathbb{R}^n$
- $\vec{v}$: a direction vector in $\mathbb{R}^n$

Then the line is:

$$L = \{\vec{p} + t \cdot \vec{v} \mid t \in \mathbb{R}\}$$

This works whether you're in:

- $\mathbb{R}^4$: spacetime 🕰️
- $\mathbb{R}^{1000}$: data science, AI embeddings 📊🤖
- $\mathbb{R}^\infty$: function spaces, advanced physics 🌐

Even though you can't **visualize** these spaces, **linear algebra lets you operate confidently in them** 💪✨

🧠🔑 Key Takeaways

- A **parametric equation** of a line has the form:

  $$\vec{x}(t) = \vec{p} + t \cdot \vec{v}$$

  where:

  - $\vec{p}$ = position vector (a point on the line)
  - $\vec{v}$ = direction vector (the line's slope or orientation)
  - $t \in \mathbb{R}$ = scalar parameter (like time ⏳)
- In $\mathbb{R}^2$, this approach **recovers slope-intercept lines** and much more

- In $\mathbb{R}^3$ and higher:
  - Scalar equations like $ax + by + cz = d$ define **planes**, not lines ❌
  - Parametric form is the **only way** to represent lines ✅
- This technique **generalizes to any dimension**—essential for:
  - Geometry in higher-dimensional spaces
  - Simulations in physics and graphics
  - Neural network embeddings and manifold learning