Video game character creation is a moduli problem

C Eric Overton-Walker

Abstract

In this talk, I want to give an introduction to the concept of a moduli problem, which in general asks us to find a space that parametrizes solutions to a given problem. I will give examples of all skill levels, including video game character creation, high school geometry, introductory topology, conic sections, algebraic geometry, and beyond. My hope is to show you that the language of moduli problems is a fun and interesting tool!

The big picture, motivated by a small example

Let's get the clickbait title out of the way right now: "Video game character creation is a moduli problem." What do we mean by video game character creation, by moduli problem, and in what way are they the same?

Lots of video games let you customize your own or other characters during the game. Think the Sims, Oblivion, Dark Souls, VR Chat, Spore, etc. To customize your character, you're often given a screen with a list of characteristics and can see how prescribing different attributes to those characteristics changes the character you're designing.

Let's think about a simple example of a character creator with only two characteristics: height and weight. Perhaps this is presented to the player in the form of sliders: the player moves one slider left and right to make the character shorter or taller, and a second slider left or right to make the character thinner or fatter.

These sliders, in terms of the logic of the game, are just mathematical intervals; perhaps we normalize them to both be $[0, 1]$. Thus a choice of a character's attributes is a choice of a height $h \in [0, 1]$ and a weight $w \in [0, 1]$. The "space" of all such characters you can create is in bijection with $(h, w) \in I^2$.

Not only do the points in I^2 correspond to the characters you can create, but we also have a notion of what it means when two characters are similar. Intuitively, we know that moving the sliders very slightly produces characters with roughly the same heights and weights (character creation is illogical if this is not the case!) and this is reflected in the space by the proximity of $(h \pm \varepsilon_1, w \pm \varepsilon_2)$ to (h, w).

This is the core idea of a *moduli problem*.

Definition 1. A **moduli space** (synonyms: classifying space, parametrizing space) is a space whose points parametrize solutions to some (often algebro-geometric) problem. A **moduli problem** asks us to find a moduli space, if one exists, for a given problem. We then use properties of the moduli space (open neighborhoods or boundaries, for instance) to understand solutions to the moduli problem.

Under this lens, the moduli problem of determining a character's height and weight has solution I^2. The small open neighborhoods of I^2 correspond to characters which are "similar," in the sense that their heights and weights do not differ dramatically. The boundaries of I^2 correspond to the tallest/shortest and thinnest/fattest characters the game permits us to create.

We can better appreciate character creation by understanding how different changes to how you can determine your character change the resulting moduli space. If we add a third independent slider, e.g., hair length, the space becomes I^3. If we add a characteristic chosen from a discrete list, like eye color (brown or green or blue), then that corresponds to disjoint union $I^2 \sqcup I^2 \sqcup I^2$ in the moduli space. If two characteristics are codependent – say, changing a character's height constrains the allowed weights – then we have some interesting subspace of I^2.

The rest of this talk, now that you've been suitably hooked in by the clickbait, is just going to be exploring different mathematical situations, phrasing them as moduli problems, and attempting to find the moduli space solutions. Here we go!

High school geometry

Here is a cool question an inquisitive beginning geometry student might ask:

Question 1. What are all of the triangles?

Of course this question as posed is kinda nonsense, so let's refine it slightly:

Question 2. What are all of the triangles, up to similarity?

This is often the language that moduli problems are stated in. We want to classify all such objects of a given kind or with a certain property, but of course *all* such objects is usually more than what we're after. So we impose a suitable notion of equivalence: similarity, isomorphic, congruent, etc.

Solving this moduli problem means producing a space, if one exists, whose points are in bijection with similarity classes of triangles. Let's discuss how to do this.

Similarity means we can move, rotate, and rescale triangles. A triangle has at least one shortest side; let's rescale it to be length 1, and move it to have endpoints $(-1/2, 0)$ and $(1/2, 0)$ in $\mathbf{R}^2$. This has fixed two vertices of our triangle, and the third point $(x, y) \in \mathbf{R}^2$ will determine our triangle. But, not every remaining point $(x, y) \in \mathbf{R}^2$ is a point in our moduli space.

First, we rescaled so that this edge along the x-axis is the shortest edge. That means, if we draw circles of radius 1 around $(-1/2, 0)$ and $(1/2, 0)$, the points in the interiors of these circles cannot be in our moduli space, else we have produced a side shorter than length 1.

Similarly, by similarity, any point below the x-axis $(x, -y)$ produces a triangle that is similar to the triangle produced by (x, y), so we can exclude everything below the x-axis. Symmetrically, triangles given by $(-x, y)$ are similar to (x, y), and thus we may exclude the second quadrant as well.

Thus, our moduli space is the first quadrant, except for the values within distance 1 of $(1/2, 0)$. Call this space $\mathcal{M}_\Delta$. When you pick two points close to each other in $\mathcal{M}_\Delta$, you will get triangles that look similar (for the plain-english use of the word similar, not mathematically similar!).

Also, for the boundaries, notice that the points $(0, y)$ on the y-axis (which we must include!) correspond to isosceles triangles. The point $(0, \sqrt{3}/2)$ is the unique equilateral triangle up to similarity, and the points on the boundary of the circle are also isosceles triangles (those with two sides of length 1, but by rescaling, those iscosceles triangles whose equal length sides are shorter than the third, versus the y-axis, whose equal length sides are longer than the third). Additionally, the x-axis corresponds to three colinear points, which high school students probably wouldn't classify as triangles at all, though perhaps we call them "degenerate" triangles.

Finally, even though they aren't in general on the boundary, we can identify classes of triangles as subregions of $\mathcal{M}_\Delta$. The right triangles are in bijection with $\mathcal{M}_\Delta \cap \{(1/2, y) \mid y \in \mathbf{R}\}$. The acute triangles are to the left of this line and the obtuse are to the right.

Introductory topology

Next, perhaps a question whose answer you already know or will learn in your topology course, but stated using our language of moduli problems:

Question 3. What is the moduli space that corresponds to all lines through the origin in $\mathbf{R}^n$, $n \geq 2$?

Let's focus on $n = 2$; the generalization follows *mutatis mutandis.* There are several ways you might try to distinguish lines through the origin in $\mathbf{R}^2$ – perhaps you look to the angle the line makes with the x-axis. If you do this, make sure that you recognize that you need to identify π radians with 0 radians. Or you could classify the lines by their slope, but make sure that you don't miss the vertical line $x = 0$, so include "∞" slope, and identify it with $-\infty$ as well. In either case, you are writing

$$\{\theta \in [0, 2\pi) \mid \theta_1 \sim \theta_2 \bmod \pi\}$$

or

$$\{x \in \mathbf{R} \cup \{\infty\} \mid \text{open intervals containing } \infty \text{ contain all arbitrarily large positive and negative values}\},$$

but you are writing S^1, the circle. However, we want to recognize the topology in a slightly more sophisticated way which generalizes to $\mathbf{R}^n$, because in general lines through the origin don't necessarily correspond to S^{n-1}, and angle and slope become harder to define.

So, recognize that a line through $(0, 0)$ is composed of points (x, y) such that for any (x, y) on that line, $(\lambda x, \lambda y)$ is also on that line. So for any given line, it's enough to tell me one point on it, as long as that point isn't $(0, 0)$ itself, for which all lines pass through. Thus, our moduli space, while it is S^1, is also

$$\{(x, y) \in \mathbf{R}^2 \mid (x, y) \sim (\lambda x, \lambda y) \text{ for } \lambda \neq 0, \text{ not both of } x \text{ and } y \text{ are } 0\}.$$

This describes *homogeneous coordinates*, written $[x : y]$, and is the space $\mathbf{P}^1_{\mathbf{R}}$, projective (1-) space (over $\mathbf{R}$). In fact, in general, the moduli space that corresponds to all lines through the origin in $\mathbf{R}^{n+1}$ is

$$\mathbf{P}^n_{\mathbf{R}} = \{[x_0 : \cdots : x_n] \mid [x_0 : \cdots : x_n] \sim [\lambda x_0 : \cdots : \lambda x_n] \text{ for } \lambda \neq 0, \text{ not all } x_i \text{ are } 0\}.$$

Conic sections

Recall that a conic section is classically the intersection of a cone with a plane. We get circles, ellipses, hyperbola, parabolas, and degenerate conics like double lines and points. Moving from geometry to algebra, one can write the cone without loss of generality as $x^2 + y^2 = z^2$ and thus intersecting with a plane produces an equation which we can write as the vanishing of a generic polynomial in $\{x, y, z\}$ of degree 2 (in fact, all such conics appear this way):

$$a_0x^2 + a_1xy + a_2y^2 + a_3xz + a_4yz + a_5z^2 = 0.$$

Here of course the choice of a_i determines the conic section, and while choosing all a_i to be 0 satisfies the equation, we want to exclude this trivial case.

Question 4. What are all of the conic sections?

Note also that if we only need to consider the curve itself, then it's easy to see algebraically that the equation is invariant under scaling:

$$\lambda \left(a_0x^2 + a_1xy + a_2y^2 + a_3xz + a_4yz + a_5z^2\right) = \lambda \cdot 0 = 0.$$

But notice, this is exactly like the previous section! A homogeneous curve of degree 2 corresponds to a tuple of a_is, invariant under scaling, and not all a_i zero, and thus it corresponds to a point

$$[a_0 : a_1 : a_2 : a_3 : a_4 : a_5] \in \mathbf{P}^5!$$

Thus $\mathbf{P}^5$ is the solution to two moduli problems: lines through the origin in $\mathbf{R}^6$ and conic sections.

But what is a moduli problem, really?

We've described moduli problems generally as the goal of classifying all objects (up to some equivalence) and understanding what it means for objects to be "close to one another." This can be made a bit more precise using the language of families.

Definition 2. A **moduli problem** is

1. a class of objects $\mathcal{P}$,
2. a notion of families of objects over bases, and
3. a notion of equivalence of families.

Families play the role of objects being "close to one another." Since an object is "close to" itself, #3 gives us the equivalence relation on objects. In particular, here's what a family is:

Definition 3. A **family** of $\mathcal{P}$-objects over a base B is a space X with $\pi : X \twoheadrightarrow B$ such that $\pi^{-1}(b) := X_b$ is a $\mathcal{P}$-object. (I'll draw the picture that one must draw.)

Different bases describe different ways you want to parametrize $\mathcal{P}$-objects. Want to imagine varying $\mathcal{P}$-objects via a 2-second long movie? Let $B = [0, 2]$ and for $b \in B$, X_b is the frame at second b. Want to vary your $\mathcal{P}$-objects in a periodic way? Maybe then $B = S^1$. Et cetera.

Really though, people state moduli problems in terms of functors. This is just a notational quirk and doesn't say anything new:

Definition 4. A **moduli problem** is a functor F into **Set** that takes a base B to an equivalence class of a family $X \twoheadrightarrow B$.

Remark 1. I probably will not say this out loud during the talk, since I'm pitching this to early grad students and assuming few prerequisites, but F acts on morphisms $f : B_1 \to B_2$ by pullback: the family $X \twoheadrightarrow B_2$ is sent to $f^*(X) \twoheadrightarrow B_1$. (Thus F is contravariant.) While I'm on the topic of things I'm sweeping under the rug, notice I'm playing fast and loose with the source of F (and I really ought to be doing the same with **Set** as well). ☺

As an example, the moduli problem of "lines through the origin in $\mathbf{R}^n$" is the functor which takes a base B to a family of lines through the origin over B. A family is just a space X where X_b corresponds to a line through the origin.

A solution to a (fine) moduli problem is just showing your functor is representable – i.e., a functor of points

This is throwing a lot of language at you, but keep in mind our examples we've already seen. We've already solved the moduli problem of lines through the origin in $\mathbf{R}^n$ – it's $\mathbf{P}^{n-1}$. Now we've restated the moduli problem as the functor which takes a base B to a family of lines through the origin over B. We just want to restate the solution now. We claim the key idea is that of a functor of points:

Definition 5. A **functor of points** is a functor $\mathrm{Hom}(-, M)$ for some space M.

This is called a functor of points because, if I hand you a space M, how do you find the points of that space? You consider all possible maps from a singleton $\{*\} \to M$, or in other words, $\mathrm{Hom}(\{*\}, M)$.

Of course, you don't just have to plug in $\{*\}$ into the functor of points. You could plug in I and get $\mathrm{Hom}(I, M)$, and we know all maps $I \to M$ are just the paths in your space. Or, $\mathrm{Hom}(S^1, M)$ is all the loops in your space. In general, $\mathrm{Hom}(B, M)$ is just the "B-valued points of M."

But notice what a solution to a moduli problem is. We said it's a space M whose points correspond to the $\mathcal{P}$-objects we're classifying, and the neighborhoods of M should tell you about how related different objects are. We've described this relationship via families over a base B, and thus observe:

Let M be a moduli space.

$\mathrm{Hom}(\{*\}, M)$ is the points of your moduli space, in bijection with the $\mathcal{P}$-objects.	A family over a point $X \twoheadrightarrow \{*\}$ has a single preimage and thus a single $\mathcal{P}$-object.
$\mathrm{Hom}(B, M)$ is the B-valued subspaces of M. Subspaces correspond to relatedness of $\mathcal{P}$-objects.	A family over a general base $X \twoheadrightarrow B$ tells you how to parametrize $\mathcal{P}$-objects via B.

Thus, solving a moduli problem, i.e., giving a parametrizing space M, is the same as saying the functor F which described the moduli problem in the first place is equivalent to a functor of the form $\mathrm{Hom}(-, M)$ for that space M. Solving a moduli problem is showing that it is a functor of points!

Definition 6. When a functor is equivalent to a functor of points $\mathrm{Hom}(-, M)$, we say it is **representable** (by M).

Can you solve every moduli problem? Is every functor representable?

Nope.

Question 5. Can you find a moduli space for one-dimensional $\mathbf{R}$-vector spaces, up to isomorphism?

Well, if you could, it'd have to be a point, since the space needs to be in bijection with the objects, and there's only one one-dimensional vector space up to isomorphism.

A family of one-dimensional vector spaces over B is just a line bundle, so even though we have the more advanced language of functors in our back pocket, we can still visualize this moduli problem easily.

Consider for example, families over $B = S^1$. There are in fact not one but two line bundles over S^1! They are the cylinder and the Möbius loop. But $|\mathrm{Hom}(S^1, \{*\})| = 1 \neq 2$, and thus we cannot solve this moduli problem.

In fact, more generally, the existence of automorphisms often forces the answer to be no. (Quotients of schemes, moduli of elliptic curves, etc.) What might happen next is you impose extra hypotheses on your moduli problem which kill off all these extra automorphisms, or (if you're not a coward ☺) widen your frame of mind from "moduli spaces" to "moduli *stacks*," but that is a talk for another time.