Received: 31 July 2019 | Revised: 14 October 2019 | Accepted: 15 October 2019

DOI: 10.1002/pro.3752

TOOLS FOR PROTEIN SCIENCE

Peppy: A virtual reality environment for exploring the principles of polypeptide structure

David G. Doak[1] | **Gareth S. Denyer**[2] | **Juliet A. Gerrard**[3,4] | **Joel P. Mackay**[2] | **Jane R. Allison**[3]

[1]Games Art and Design, Norwich University of the Arts, Norwich, UK

[2]School of Life and Environmental Sciences, University of Sydney, New South Wales, Australia

[3]School of Biological Sciences, University of Auckland, Auckland, New Zealand

[4]School of Chemical Sciences, University of Auckland, Auckland, New Zealand

Correspondence
David G. Doak, Norwich University of the Arts, Norwich, NR2 4SN, UK.
Email: david@ddoak.com

Gareth S. Denyer, School of Life and Environmental Sciences, University of Sydney, NSW 2006 Australia.
Email: gareth.denyer@sydney.edu.au

Jane Allison, School of Biological Sciences, University of Auckland, Auckland 1010, New Zealand.
Email: j.allison@auckland.ac.nz

Funding information
Rutherford Discovery Fellowship, Grant/Award Number: 15-MAU-001; School of Chemical Sciences, University of Auckland; University of Auckland Vice-Chancellor's Strategic Development Fund

ABSTRACT

A key learning outcome for undergraduate biochemistry classes is a thorough understanding of the principles of protein structure. Traditional approaches to teaching this material, which include two-dimensional (2D) images on paper, physical molecular modeling kits, and projections of 3D structures into 2D, are unable to fully capture the dynamic 3D nature of proteins. We have built a virtual reality application, Peppy, aimed at facilitating teaching of the principles of protein secondary structure. Rather than attempt to model molecules with the same fidelity to the underlying physical chemistry as existing, research-oriented molecular modelling approaches, we took the more straightforward approach of harnessing the Unity video game physics engine. Indeed, the simplicity and limitations of our model are strengths in a teaching context, provoking questions and thus deeper understanding. Peppy allows exploration of the relative effects of hydrogen bonding (and electrostatic interactions more generally), backbone φ/ψ angles, basic chemical structure, and steric effects on a polypeptide structure in an accessible format that is novel, dynamic, and fun to use. Apart from describing the implementation and use of Peppy, we discuss the outcomes of deploying Peppy in undergraduate biochemistry courses.

KEYWORDS

polypeptide, protein, secondary structure, teaching, undergraduate, virtual reality

1 | INTRODUCTION

The principles of protein structures are the threshold learning outcomes for fundamental undergraduate biochemistry courses. Understanding the structures and conformational preferences of amino acids, and their capacity to achieve nonbonded interactions such as hydrogen bonds and ion pairs, is of fundamental importance for the formation of regular secondary structure elements such as α-helices and β-sheets. Functional competence with protein structure requires not only committing these principles to memory but also gaining a sense of how a protein's three-dimensional (3D) structure emerges from the interplay of the underlying physical and chemical characteristics.

Traditional approaches to teaching students about protein structure include textbook 2D images on paper through to physical molecular modeling kits and to projections of 3D structures into 2D such as stereograms and protein molecular graphics programs. Although these are all useful in different ways, all suffer from limitations.

Proteins are inherently 3D objects, and thus any 2D representation will fail to provide a complete picture. Physical models are 3D, but are fragile and time consuming to assemble and disassemble. Moreover, the behavior of proteins is intrinsically dynamic and results from the interplay of a host of physicochemical forces, which are difficult or impossible to represent in rigid models or on paper. 3D molecular visualization software packages are typically only used in undergraduate classes at second year and above, and even these tools are generally limited to observing predetermined static structures and still require the mind to derive a 3D understanding from a 2D computer screen.

For these reasons, an alternative approach that might assist students in understanding the underlying principles is to use a virtual reality (VR) environment. VR is intrinsically 3D and allows both representation of dynamic behavior and "hands-on" manipulation by the user. The potential benefits of teaching protein structure using an interactive VR approach have already been reported.[1–4] The particular strengths of VR are not limited to its novelty or connection to gaming, but are derived from the physical involvement and visual immersion of the user, which is facilitated by having a head-mounted display and the availability of six degrees of freedom (6DoF; translation along and rotation about each of three orthogonal axes).

There are many existing examples of the use of VR technology to visualize experimental data[5,6] and facilitate the investigation of cellular[5] (e.g., http://thebodyvr.com) and molecular[3,7–17] (also e.g., http://nanome.ai, https://gwydion.co, https://research.nanosimbox.io) scenarios. Most, however, are targeted at researchers rather than undergraduate students and tend to illustrate known protein structures rather than encourage their production. Although this does not prevent their use in teaching, with real-time interactive molecular dynamics simulations in VR being successfully used to teach enzymatic catalysis,[4] the design goals of a tool aimed at teaching are quite different from those of a tool aimed at researchers. Effective teaching requires a fun and intuitive environment that encourages self-directed and creative engagement and leads the students to ask questions; thus, a degree of fallibility is desirable and genuine exploration and productive failure are essential.

Although the case for using VR in teaching molecular processes is compelling, the impetus to create specialized applications is somewhat reduced by the fact that deployment of VR to large undergraduate classes is limited by the difficulty in providing sufficient equipment for typical undergraduate class sizes. Furthermore, even when suitable software exists, deployment requires agility in course management to allow rapid introduction into the curriculum. We note, however, that through careful organization, class sizes of several hundreds of students can make use of VR facilities comprising 25–30 headsets.

We present here a VR tool, "Peppy," aimed at facilitating the teaching of the principles of protein structure to undergraduate classes. Peppy allows exploration of the relative effects of hydrogen bonding (and electrostatic interactions more generally), backbone φ/ψ angles, basic chemical structure, and steric effects on the resulting polypeptide structure. Additionally, we describe the prototyping of Peppy in undergraduate biochemistry courses at the University of Sydney, which possesses a dedicated VR facility, the Immersive Learning Laboratory (ImmLL). Development of Peppy and the existence of the ImmLL, along with careful yet adventurous course design and management, overcame the aforementioned issues with deploying VR in teaching.

2 | RESULTS

We first outline the principles and goals underlying our approach and then describe the high-level functionality of and user's interaction with Peppy, followed by details and outcomes of its deployment in undergraduate biochemistry classes. A detailed description of the implementation is provided in Section 4. Although both our implementation and our testing to date are preliminary, we think Peppy shows great promise as a teaching tool. The code is publicly available via GitHub, and we encourage others to try it out and provide us with feedback that we will harness to inform further development.

2.1 | Underlying principles and goals

Our goal was to create an environment that allowed students to engage with protein structure and dynamics and gain an understanding of how these are determined from the underlying physical and chemical features of polypeptides. Our fundamental philosophy was to encourage experiential learning about the conformational properties of polypeptides through play.

We have harnessed the gaming associations of VR, as well as its ability to provide 3D information at a human scale, to enhance student engagement and "make learning fun." To facilitate understanding and experimentation, absolute physical/chemical correctness is less important than usability. We therefore allow interactive alteration of just the major factors that influence secondary and tertiary structures. To maximize simplicity, the tunable factors are limited to those that have most impact on protein secondary and tertiary structures, namely

residue types, φ/ψ angle values, hydrogen bonding, and electrostatic interactions. The relative strengths of the latter three factors can be adjusted by the user.

2.2 | High-level functionality

Peppy allows the user to create polypeptide chains that can then be "physically" grabbed and manipulated in the virtual space. By pushing, pulling, twisting, and "touching" these molecules, higher order structures can be created or destroyed and their stability and properties investigated. The minimal game-like environment encourages self-directed creative engagement. Interaction is immediate and intuitive and is built on both the immersive nature of VR and the revolutionary interface possibilities afforded by fully tracked 6DoF motion controllers. Many of the low-level simulation parameters (e.g., force constants, dynamics) are exposed to the user and can be manipulated directly and the consequential effects are observed. Peppy is not intended to be a robust and detailed molecular dynamics simulation; however, it is highly effective as a representative sketch that allows the user to explore many of the emergent structural properties of proteins such as repeating secondary structural elements. Hydrogen bonds and electrostatic interactions are modeled in a simplified manner and are represented graphically by animated particle effects that visualize the dynamic forces involved.

Backbone (φ and ψ) dihedrals are visualized on an interactive Ramachandran plot and can be manipulated and monitored by the user. Amino acid sequences can be easily altered in order to visualize and investigate the impact of side-chain conformations and steric properties. An in-game camera allows the user to record snapshots of their creations in association with an avatar that projects their physical presence within the virtual environment.

It is also possible to run the application in a "flat" non-VR mode—this presents the same innovative dynamic functionality but is limited by a more traditional mouse/keyboard interface. While losing the immersive nature of the VR environment, this mode allows students to explore the application on a standard Windows operating system without the requirement for VR hardware, thus broadening the penetration of the software.

2.3 | User interaction

Peppy includes a variety of methods for the user to interactively manipulate peptide molecules in real time. Most interactions can be controlled and reported though the main dashboard, which is shown in Figure 1. Information about Peppy, including a video illustrating its use, is available at https://sydney.edu.au/science/our-research/research-areas/life-and-environmental-sciences/peppy-vr.html.

In a standard interaction, a user first builds a peptide backbone of a particular length. They may then manipulate the torsional angles between the peptide units to create motifs of secondary structures either through manual manipulation or by selecting the areas on the Ramachandran plot. In addition, users may explore the influence of particular side-chain residues by altering (or "mutating") the amino acid sequence.

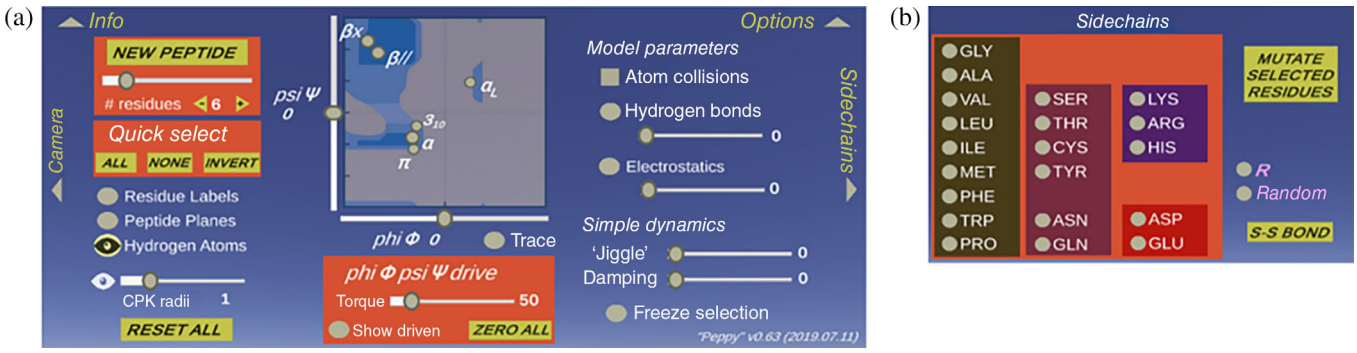

FIGURE 1 The Peppy interaction dashboard. (a) Main dashboard and (b) pop-out side-chain menu. Arrows near the edges can be "clicked" to open out additional specialist menus. The *tunable parameters*, which are adjusted for selected residues using sliders, are hydrogen bond strength, electrostatic interaction strength, φ/ψ angle values, φ/ψ drive torque, visualization of atomic radii, degree of "jiggle" dynamics, and damping of the dynamics. The *binary options*, which are turned on or off using radio buttons, are the calculation of forces because of atom collisions, the visibility of hydrogen bonds, electrostatic interactions, peptide planes, and hydrogen atoms, the freezing of selected residues, and the illustration of the φ/ψ traces, highlighting the residues with driven φ/ψ and the numbering of residues in the Ramachandran plot. *Adjustable peptide properties* are the peptide size (number of amino acids) and the type of each amino acid type (via the pop-out *side-chain* menu)

2.4 | Building a peptide

First, when constructing a polypeptide unit, the user chooses the number of amino acids and a peptide with "dummy" (R) side chains appears. These R groups can subsequently be mutated individually or in groups to any amino acid type. There is also an option to randomly assign the amino acid type. Residue numbers are shown if the "Residue Labels" radio button is selected. Visibility of peptide planes and hydrogen atoms can also be controlled in the same manner.

2.5 | Selection

Individual residues can be selected by using a controller to point at and "click" on any of the atoms. Contiguous atoms can be selected as a group, and all the residues in a peptide can be selected or deselected through buttons in the "Quick Select" box. Once selected, the type of amino acid and the φ/ψ angles can be modified though the dashboard.

The peptide(s) and the menu can be moved closer or further away by remote interaction using the "tractor beam." Peptides can be "grabbed" directly through the user's virtual hands or "remotely grabbed" and directly pushed, pulled, or tangentially dragged using the tractor beam. The remote grabbing interaction is sufficiently intuitive and direct to allow the user to carry out sophisticated manipulations. Remarkably, the flexibility and smoothness of the molecular handling is sufficient to allow a peptide backbone to be tied in a knot using a single motion controller.

2.6 | Physical effects

The contribution of steric effects can be switched on or off using the "Atom Collisions" radio button. The effective size (radius) of each atom is fixed, but atoms can be visualised at different sizes by tuning atom size using the Corey-Pauling-Koltun (CPK) radii slider.

The φ/ψ angle values are visualized on a 2D map modeled on the Ramachandran plot commonly used to analyze protein structure. Major secondary structure types are represented by selectable spots on the map. The φ/ψ angle values of a single residue or group of selected residues can be controlled by clicking on a position in the 2D map or by the sliders on each axis of the map. The sliders direct the parameters of the configurable joints representing the φ/ψ bonds. The strength by which φ/ψ angle values are driven to the selected values is controlled using a slider, which adjusts the apparent torque of the bonds.

The current φ/ψ angle values of each residue are dynamically displayed. Driven dihedrals can be visualized via a toggle switch, and the history of the φ/ψ angle values can be shown by selecting the "trace" option.

2.7 | Chemical properties

Atoms are colored by a type similar to commonly used atom rendering standards, with carbon black, nitrogen blue, oxygen red, hydrogen white, sulfur yellow, and "dummy" side chains magenta; the atomic radii depend on the atom type.

Although hydrogen bonds are a subset of electrostatic interactions, in Peppy they are modeled explicitly to allow students to explore their effects on proteins' secondary and tertiary structures, independent of other types of electrostatic interactions. Both electrostatic interactions and hydrogen bonds can be visualized independently by checking their respective radio buttons. Similarly, the strength of both types of interactions can be scaled using a slider.

2.8 | Dynamics

An essential part of Peppy is that the simulation is interactive and dynamically responsive in real time. By default, the peptide is static, but any forces added by the user (e.g., by grabbing and manipulating part or all of a peptide) cause a wave of movement through all the residues in response. The resultant forces are heavily damped to prevent the occurrence of anomalously large velocities.

Random motion, in which each *rigidbody* (see Section 4) experiences impulse forces of random direction that modify the inherited momentum, can be incorporated by switching on "jiggle." This is notionally equivalent to thermal motion, but no attempt is made to correlate the scale of the jiggle dynamics with a particular macroscopic temperature. The dynamics are also damped by scaling the translational and rotational motions. Both the size of the impulse forces and the degree of damping can be adjusted.

Selected residues may be "frozen" to allow the user to easily lock the conformations of the polypeptide regions to facilitate more directed manipulation elsewhere.

2.9 | Outcomes and deployment

We deployed Peppy in undergraduate biochemistry classes at the University of Sydney in August 2018 and March 2019. The University of Sydney's ImmLL has 26 high-powered

PCs (Intel i7-7700 3.6 GHz, 16 GB RAM, NVIDIA GTX1070) and Oculus Rift VR headsets. The first cohort comprised a class of 80 biochemistry students (course code BCHM3082), the majority of whom were in their final semester at the University. Students worked in pairs, allowing the entire cohort to trial Peppy in two sessions and thus overcoming the discrepancy between the class size and the number of VR headsets. There was no assessment component; rather, we aimed to investigate user's acceptance and capability, as well as the logistics of running large-class VR sessions in a dedicated facility. Interestingly, less than 5% of the cohort reported ever having worn a VR headset, but this did not appear to be an impediment to the majority of students. Only two students felt that their eyesight impeded their engagement. While some students felt slightly nauseous after an extended period wearing the goggles, less than 1% reported feeling overtly nauseous, which can be a common issue with VR headsets. We believe this is due to Peppy being able to be used sitting down rather than standing, as it does not require the user to move around, as well as the students working in pairs, which allowed them to swap places if they became slightly nauseous. We also found that incidences of nausea decreased when the room was less crowded and after recent improvements to the airflow.

Encouraged by the reception, we carried out several iterations of development of Peppy before using it for a class of 85 third-year students in a course specifically related to protein structure and function (BCHM3072). As with the first cohort, these students were familiar with the fundamentals of protein secondary structure, and the majority had explored this in the second-year foundational courses by constructing physical models of short polypeptides using the Cochranes' Orbit molecular

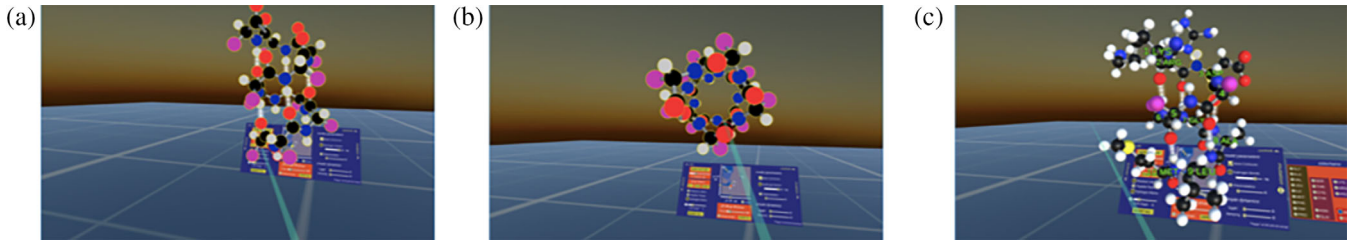

FIGURE 2 Examples of α-helices built by students. (a) Angled view showing hydrogen bond formation in an α-helix built with dummy side chains (magenta); (b) end-on view showing the positions of the side chains for the same α-helix; (c) an α-helix with post hoc specification of the amino acid side chains so that the uppermost residues are hydrophilic and the lower residues are hydrophobic. The students commented "We initially tried to put extremely large residues in the helix like tryptophan, but we soon found out that this made the helix unwieldy. We therefore largely stuck to smaller residues like glycine, lysine, and alanine. The R groups of the top section began to come close to ionically bond together as we put an aspartate and lysine/arginine group next to each other. We did this deliberately to see how rigid the helix was, and whether the ionic strength would pull apart the helix"

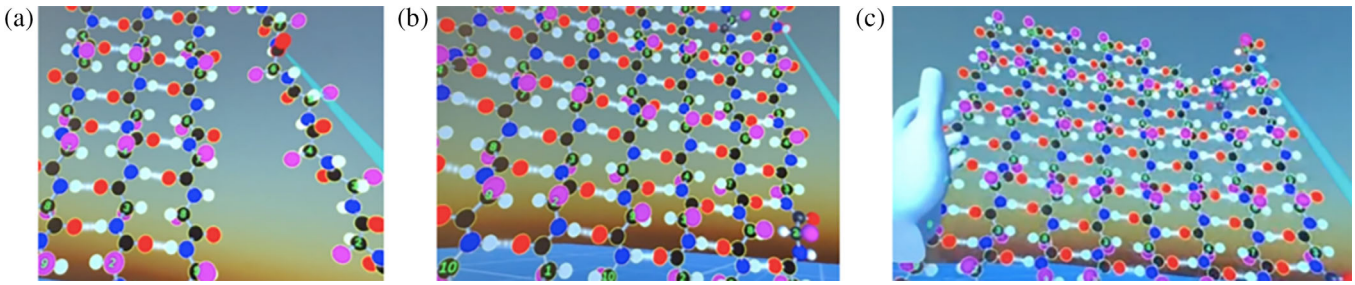

FIGURE 3 Example of a β-barrel built by a student. (a) The student has constructed an antiparallel β-sheet using three decapeptides. A fourth is being brought in from the right. This quickly snaps into place from top to bottom with a zip-like smoothness, as the locked hydrogen bonds on the main sheet direct the conformation of the incoming peptide. (b) After four subsequent decapeptides have been added, an eight-peptide sheet is formed. It has taken <10 min to build this structure. (c) Now, the student is faced with trying to fold the entire structure in on itself so that a barrel can be formed. This proved to be too difficult—"like trying to fold a bedsheet in the wind." However, the student learned a valuable lesson about both the strength and flexibility of the sheet. In addition to stimulating discussion about other ways to complete the task, this experiment also gave us the impetus to incorporate the "freezing" function into Peppy

building system (https://www.cochranes.co.uk/). Despite this, their depth of understanding of many concepts that we consider necessary for fluency when discussing structural biology is largely still developing.

Students worked in pairs for about 90 min to complete a series of tasks using Peppy. The workflow was intended to take the students from the creation of short peptides, for the revision and exploration of basic peptide bond geometry, through to the construction of complex polypeptides, and the formation of hydrogen-bonded secondary structures (α-helices and β-sheets). Students were encouraged to investigate the effects of changing tunable-parameter values and amino acid side chains to see how this affected the secondary structure elements that they had made (Figure 2).

Anecdotally, we observed that working in pairs was exceptionally effective, with students taking turns to wear the headset or read and interpret the written instructions. This alternation of pilot and navigator fostered engagement, reduced the burden of wearing the headset for extended periods, and created a strong culture of interstudent support and desire to achieve mastery. This prompted not only discussion about the features of Peppy that were exciting and could be improved, but also led students to openly confess previous misconceptions and provide explanation to one another.

After completion of the workflow, students were encouraged to explore all the functions of Peppy and, almost unanimously, they enthusiastically built extravagantly complex polypeptides, experimented with the effects of different amino acid side chains, and attempted to construct multichain tertiary structures such as β-barrels (Figure 3) and even real, small proteins (Figure 4). One student attempted to recreate the active site of trypsin by arranging and orienting key residues in 3D space, taking directions from a representation of the structure in PyMOL. As expected, this was far too ambitious but, as with all these cases, the struggle made the student appreciate the incredible complexity and beauty of these natural structures. Throughout, the selfie camera and self-avatar features proved to be effective lubricants for student engagement.

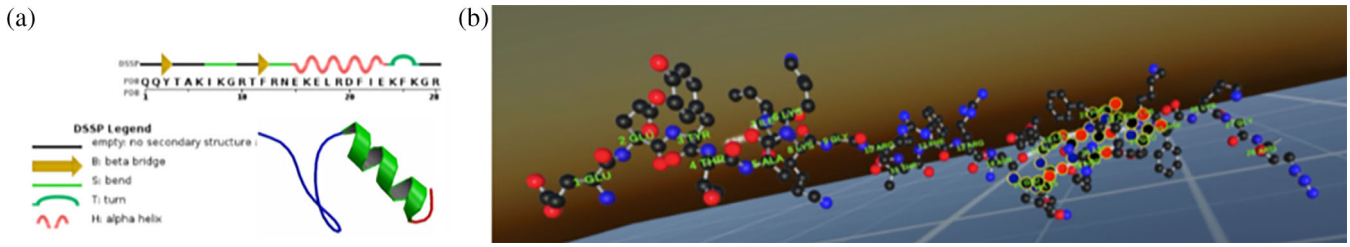

FIGURE 4 Example of a real peptide structure built by a student. (a) Sequence, secondary structure and cartoon structure of the 28-residue peptide, a section of a zinc finger (PDB ID: 1FSV). After selecting residues 15–24 and adjusting the φ and ψ angle values on the Ramachandran plot to −57° and −47°, respectively, the student observed that section to smoothly settle into an α-helical configuration, with the hydrogen bonds (white) stabilizing a rigid configuration. The student was then able to reflect on the orientation of the side chains, the space that they occupy, and the possible forces between them. Residues 1–14 did not so easily adopt a β-sheet, but this was valuable in helping the student think about what has to happen for proteins to fold. Indeed, had the "freeze" function characteristic of the latest build of Peppy been available, the task of moving these residues to form their native configuration would have been simpler

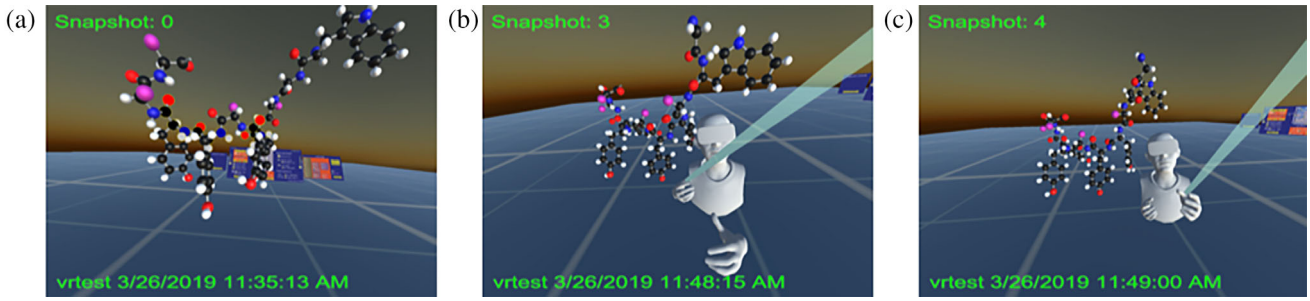

FIGURE 5 Examples of animals made by a student in the second task. (a) Initial attempt at a 10-residue peptide dog, which the student noted looks more like a giraffe due to the long neck. (b,c) Refined 10-residue peptide dog where the tryptophan head has been moved to residues 9 and 10 mutated to glycine to create an ear. (a) is a standard screen shot and (b) and (c) were taken using a selfie camera. The student noted that they liked image (c) because the dog appeared to be looking down upon their avatar. This exercise taught the student a lot about steric hindrance and the effect of φ and ψ angle values on overall structure

The students completed two assignments related to their experience with Peppy. The first was a standard laboratory report that provided proof that they had thoughtfully worked through the tasks. It took the form of screenshots embedded into a contextual narrative explaining the process, outcomes, and lessons learned. There examples of these are provided in Data S1.

The second task was more reflective and extrapolative. Students were asked to suggest workflows for the deployment of Peppy in second-year classes and were encouraged to propose the features they would like to see added or changed. Many students more than fulfilled their obligations by trying to create ambitious and/or whimsical structures (Figure 5) or perform other manipulations, which tested the limits of the software.

During both testing phases, the students provided an abundant list of desirable features and noted problems with the existing implementation, many of which we added or fixed to arrive at the version reported herein, highlighting the agility of working within a game engine framework and the advantages of including frontline teaching academics in the development team. In many cases, such as the possibility of forming multivalent hydrogen bonds, the simplicity of the model provoked discussions between and with the students that led them to challenge and ultimately deepen their understanding. This was facilitated by a favorable staff–student ratio (1:6) along with a lead demonstrator with experience in the use of Peppy and familiarity with its limitations.

3 | CONCLUSIONS

Our goal was to develop a VR application to facilitate teaching the principles of protein structure. We hypothesized that the experience of using VR would be progressive and would provide additional insight over what has previously been available, such as 2D printed images, 3D graphics projected onto 2D, and physical models. We harnessed the existing Unity video game physics engine and game development protocols to facilitate rapid prototyping and responsive development. We did not attempt to replicate the true underlying physical chemistry although we did take inspiration from existing modeling protocols such as molecular dynamics simulation force fields.

The resulting program, Peppy, presents the basic elements of protein secondary structure in an accessible format that is novel, dynamic, and extremely tangible as well as fun to use. It is possible to easily and quickly investigate a wide range of conformational properties of the polypeptide backbone and sidechains. Remarkably, despite its simplicity, it is possible to build a large variety of complex multipeptide structures using Peppy. In fact, the simplicity of the simulation becomes a strength when framed within the teaching and learning process as it provokes questions about the validity of the model. Indeed, students are specifically encouraged to test and probe the rules driving the simulation by changing the tuneable factors and exploring collisions and tensions. In all these ways, Peppy invites active challenge, questioning and critique by the students, all of which hopefully lead toward a deeper understanding of and interest in the forces and factors affecting the protein structure.

4 | METHODS

4.1 | Development strategy

Our goal, and thus our design approach, was to model a traditional physical ball-and-stick representation of molecular (peptide) structure with a dynamism that would enhance student engagement but with a realism that would ensure quality learning. To achieve this, we took advantage of the existence of video game development engines that have at their core a robust physics engine and 3D rendering, while also offering the ability to quickly prototype an application, thus allowing rapid and agile cycles of design and testing.

Achievement of our goals does not require the same degree of realism as the force fields used in molecular dynamics simulations, nor does the visual representation need to be as sophisticated as existing molecular visualization tools such as Pymol[18] and Visual Molecular Dynamics.[19] Indeed, in contrast to the latter tools, the refresh rates and rendering required for a pleasant user experience impose a further limitation on functionality.[20] Peppy does not, therefore, represent a robust, fully featured molecular dynamics simulation, but rather the simplest possible functional model of a polypeptide chain within a game engine. However, the fidelity of the physics within the game engine is very high, and the underlying computational methods are not dissimilar to those used in molecular dynamics simulations. Crucially, the end result is dynamic with an intuitive game-like interface that is highly interactive in real time.

4.2 | Implementation

Peppy was created using the Unity game engine (https://unity3d.com). We note that the units are those used in Unity and are in general at a human scale, for example, distances are in meters and weights are in kilograms, as is the standard practice in game development. Geometry and prefabricated (*prefab*; see Section 4.3) components were created within the Unity editor and the associated

code is written in C#. Some code components are licensed using the Oculus Software Development Kit (SDK). The source files and compiled executables for Peppy are available at https://github.com/ddoak/peppy.

Peppy runs on any VR-capable desktop machine with Oculus Rift headsets and touch controllers, and it is also available for Oculus Quest. This does, however, create some restrictions on the system size that can be handled without substantially affecting performance with a soft limit of 32 residues. Four 8-residue peptides will be handled better than one 32-residue peptide, especially if only one is active at any point in time, as there are only intrapeptide physics-based constraints. To further broaden its accessibility, it may also be run in a non-VR "flat screen" mode without Oculus hardware. In this mode, the user's movement and interaction are controlled using a mouse and a keyboard.

Peppy describes the polypeptide chain at an all-atom level of detail, in keeping with standard molecular dynamics force fields. This representation is functionally implemented using Prefab GameObjects within Unity. In this framework, a *prefab* is a user-defined reusable template comprising a hierarchical collection of components such as transforms, mesh renderers, rigidbodies, colliders, and C# scripts, which define bespoke behaviors and properties. *Transforms* define the position, rotation, and scale of an object; *mesh renderers* render the object in 3D at the position defined by the transform; *rigidbodies* are internally rigid objects that behave according to the laws of physics; and *colliders* define the shape of an object for the purposes of physical collisions. *Configurable joints* connect the rigidbodies.

4.3 | Degrees of freedom

Polypeptides in Peppy are built from backbone and sidechain fragments, which are described in more detail later, using a united-atom representation. Backbone fragments are prefab units that comprise a single rigidbody made of two to three atoms (Figure 6). Groups of atoms are unified in this way both for computational simplicity and to focus students' attention toward the conformational changes that are most important for secondary structure. The divisions between the groups of unified atoms are generally located on rotatable bonds. The masses of all prefab unit rigidbodies are scaled appropriately to represent the combined mass of their constituent atoms (Table S1).

Atoms have individual spherical meshes for rendering in addition to fixed-radius hard spherical colliders that prevent interpenetration. The collider radii are derived from standard van der Waals radii[21] and are not adjusted for different chemical environments (Table S1).

Molecular dynamics force fields commonly exclude van der Waals interactions between bonded atoms. In keeping with this approach, colliders within a prefab unit are combined into unified colliders, and collisions between adjacent prefab units connected by a configurable joint are explicitly switched off. All other colliders are permitted to intersect, with each prefab unit experiencing the combined effect of the colliders in other non-adjacent prefab units. These interprefab collisions can be switched on and off by the user to allow exploration of the restrictions on conformation imposed by steric hindrance.

Bond lengths and angles within each rigidbody fragment are fixed (Table S2). Additionally, within the sidechains, departures from idealized sp^2 (trigonal) or sp^3 (tetrahedral) geometry are specified explicitly (Table S5).

Rigidbodies (and hence the prefab units built from them) are connected by configurable joints between the anchor points coincident with the bonded atom centers (Tables S3, S4). These represent the polypeptide backbone covalent bonds and have only one permitted DoF (axial rotation). This DoF is fixed for the peptide bond

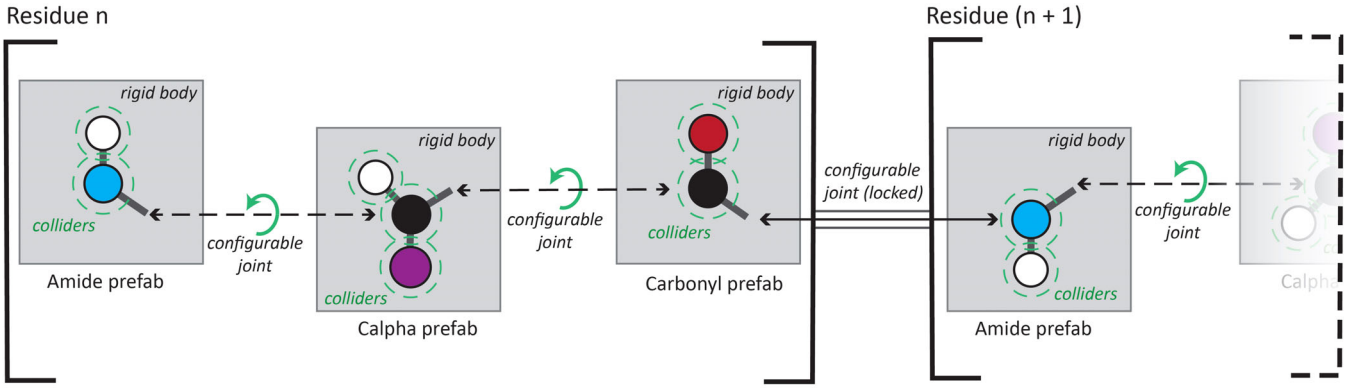

FIGURE 6 Illustration of the Peppy backbone architecture. The *n*th amino acid comprises at least three rigidbody prefab units, representing the amide (N—H) and carbonyl (C═O) functional groups and the C-alpha plus side chain (Cα + "R") unit. The units are connected by configurable joints, but only those within a residue are freely rotatable (green arrows). Each atom is represented by a collider with a fixed radius specific to that atom type (green dashed lines), where type refers to the element but not its chemical environment

but free to rotate for the central bonds of the φ and ψ dihedral angles, providing the minimum required rotational DoF (two rotations per residue).

Each rotatable configurable joint has a target dihedral angle value and an associated spring force (torque). Both can be controlled by the user. Target dihedral angle values can be chosen from a Ramachandran plot for all or selected amino acids. If the torque is nonzero, the dihedral is driven toward the target value. The torque values are not representative of real intramolecular forces; rather, they allow the user to manipulate the polypeptide backbone toward particular conformations. The scaling for the torque is empirical and was tuned during development to give a range of values that allow the user to explore secondary structure and steric hindrance.

4.4 | Rendering

The radii of the visible atomic render mesh spheres are scalable, which allows the user to transition smoothly between ball and stick representation and CPK shell representation. We note however that the collider radii do not change, only the rendering.

Rendered bonds (grey cylinders) are entirely cosmetic—they imply a fixed cylindrical mesh geometry connecting atoms. If the configurable joints are highly strained (e.g., if the user pulls the polypeptide backbone apart), then the rendered bond cylinder may no longer correspond to the actual bond.

4.5 | Amino acid architecture

The polypeptide backbone is built from three types of prefab units—N—H, H—Cα—R, and C═O—with fixed internal bond lengths and angles (Figure 6). Amino acid residues are by default created with single-atom dummy "R" side chains but may be selectively mutated at runtime to any of the standard 20 amino acid sidechains. Inter-residue disulfide bonds may also be created by selecting pairs of cysteine residues to join. Specifying a particular amino acid sidechain replaces the generic R (magenta sphere) of the backbone (H—Cα—R) with appropriate connected prefab units (e.g., CH_2, NH_3^+, OH). Key parameters describing the side chain architecture are provided in Table S5.

4.6 | Hydrogen bonds

Hydrogen bonds between backbone donor atoms (H—N) and acceptor atoms (O═C) are modeled explicitly so that they can be tuned independently of other electrostatic interactions, thus enabling students to explore their effect on protein secondary structures. Intrachain hydrogen bonds between residues i and $i \pm n$, where $n \leq 2$, are explicitly excluded. Hydrogen bonds involving sidechain atoms are currently not modeled, as these are less important for protein secondary structures.

Candidate hydrogen bonds are identified using a "spherecast" test, which sweeps a cylinder away from the donor atom, extending along the direction of the amide bond, to search for acceptor atoms (O═C). This test is interrupted by the presence of other atoms to prevent tunneling. The radius (c_r, 0.05 m) and maximum length (c_l, 0.3 m, measured between the center of the donor hydrogen atom and the nearest edge of the oxygen atom collider) of the test cylinder were empirically tuned to facilitate generation of the predicted hydrogen bonds in regular secondary structure elements. A hydrogen bond is modeled if an oxygen atom is found within $c_l = 0.3$ m of the center of a donor hydrogen atom; that is, $r_{hb} = \text{distance}(O_{collider}:H_{center}) \leq c_l$.

Hydrogen bonds are modeled with three spring joints[22] (Figure 7) to encourage linearity of the four atoms involved (N—H ... O═C). The inner spring joint connecting the hydrogen atom to the oxygen atom is modeled with a flat-bottomed attractive harmonic potential that only takes effect if the distance between the centers of the two atoms, r_{HO}, is larger than $r_{min} = 0.16$ m, and the outer spring joints connecting the nitrogen atom to the oxygen atom and the hydrogen atom to the carbon atom immediately bonded to the oxygen atom are modeled with flat-bottomed repulsive harmonic potentials that only take effect if the distance between the centers of the two atoms, r_{NO} and r_{HC}, is less than $r_{max} = 0.32$ m. Essentially, the springs attract the hydrogen atom to the oxygen atom while repelling the nitrogen atom from the carbon atom, with an overall effect of favoring a linear hydrogen bond.

The springs obey Hooke's law, with the force proportional to the displacement from the target length. This would result in extremely large forces at large interatomic distances, which would be unrealistic due to the $1/r$ distance dependence of the electrostatic forces that underlie hydrogen bonding and solvent screening effects, but do have the advantage of being robust to user's interaction. We therefore use a simple linear switching function to adjust r_{hb}:

$$r_{hb'} = \left(\frac{c_l - r_{hb}}{c_l}\right),$$

which makes the spring proportionally weaker as the donor and acceptor atoms/groups move further apart and so prevents the spring forces from becoming too large

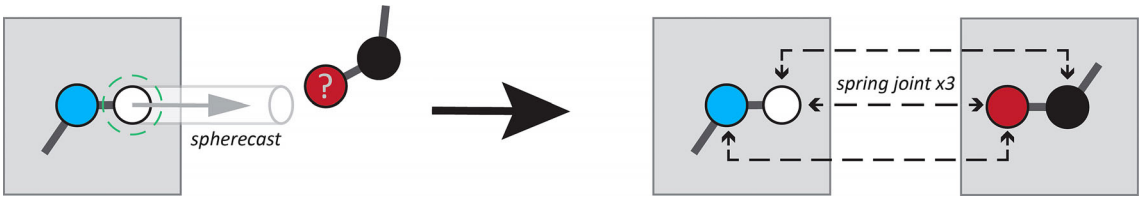

FIGURE 7 Illustration of hydrogen bond modeling in Peppy. Hydrogen bonding pairs are discovered by projecting in real time a cylinder from each donor group in line with the backbone N—H bond. If an acceptor is found, a hydrogen bond is modeled using three spring joints. The parameters of the cylinder and spring function are provided in Table S6

relative to the other forces in the model when they are stretched. The harmonic springs are also damped proportional to the velocity of the oscillation.

Taking all of these factors into account, the attractive hydrogen bonding force is calculated as:

$$F_{\text{hb,attr}} = \begin{array}{c} 0 \text{ if } r_{\text{HO}} < r_{\max} \\ -k_{\text{hb}}(r_{\text{hb}'})^2 - d_{\text{hb}}\dfrac{dr_{\text{hb}}}{dt} \text{ otherwise,} \end{array}$$

and the repulsive hydrogen bonding force is calculated as:

$$F_{\text{hb,repul}} = \begin{array}{c} -k_{\text{hb}}(r_{\text{hb}'})^2 - d_{\text{hb}}\dfrac{dr_{\text{hb}}}{dt} \text{ if } r_{\text{NO,RC}} < r_{\min} \\ 0 \text{ otherwise} \end{array},$$

where k_{hb} is the hydrogen bond spring force constant and d_{hb} is the damping coefficient. The values of $r_{\min}$, $r_{\max}$, and the force and damping constants are provided in Table S6.

Active hydrogen bonds are visualized through a simple animated particle effect that is continuously updated to be oriented directly from the H—N donor toward the O═C acceptor.

The scale for the spring forces is completely empirical and can be adjusted by the user. The lower end of the scale represents no hydrogen bond formation, while the upper end of this scale represents unrealistically strong hydrogen bonds. This deliberate choice allows the user to experiment with manipulating robust secondary structure elements.

A clear limitation of this current approach is that each hydrogen bond is independent of the presence of other hydrogen bonds. Consequently, multiple donors can hydrogen bond to a single acceptor. Although this is observed in experimental structures and in more sophisticated molecular dynamics simulations, it is overly prevalent in the current version of Peppy. The independence of hydrogen bonds does not prevent the occurrence of higher-level cooperativity of hydrogen bond formation, for instance, in the "zipping" together of β-strands into a β-sheet.

4.7 | Electrostatic interactions

The electrostatic interactions between a subset of polar atoms are modeled explicitly via a simplified Coulombic potential (Figure 8). The number of atoms with partial charges is restricted in comparison with more accurate molecular dynamics simulations for performance reasons as outlined later. Partial charges, q, are assigned to backbone amide hydrogen atoms, carbonyl oxygen atoms, and the sidechain atoms of amino acids that would be ionized at physiological pH (arginine lysine, glutamate, aspartate, and histidine). The values of the partial charges (Table S7) are derived from the GROMOS 54A8 forcefield parameters.[23,24] The total electrostatic force on a partially charged atom is given by

$$F_i = \sum_{j=1,\, j\neq i}^{N\text{pc}} 0.0125 s_{\text{elec}} \left(\frac{q_i \cdot q_j}{r_{ij}^2} \right),$$

where N_{pc} is the total number of atoms with partial charges, 0.0125 is an empirical scaling factor, r_{ij} is the distance between atoms i and j, and i and j do not reside within the same prefab unit. s_{elec} is the electrostatic strength and can be adjusted by the user in the range [0,100] to investigate the contributions of electrostatic forces to peptide structure. The resultant force is applied to the parent rigidbody at the position of atom i.

Electrostatic interactions are visualized through animated particle effects. Each charged unit has a colored (red/blue) particle system that emits radially from a spherical volume around the atom. The number, size, and acceleration of the particles are scaled according to the magnitude and direction of the current resultant electrostatic force on the atom. This effect is deliberately theatrical and is intended to be arresting for teaching purposes.

In order to contain the computational cost, the electrostatic interactions are computed and the resulting forces applied at lower frequency than the game physics (10 vs. 90 iterations per second). Moreover, because the current implementation does not use neighbor lists or distance exclusions, the cost of calculating the

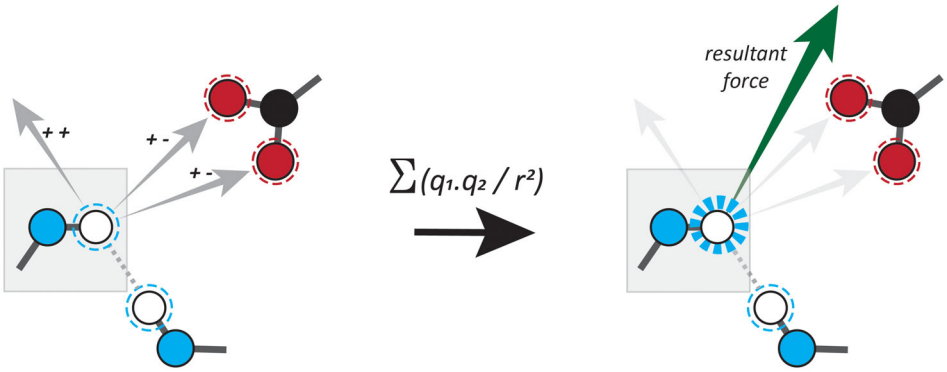

FIGURE 8 Illustration of electrostatic interaction modeling in Peppy. The total force on a partially charged atom is the sum of its Coulombic interactions with all other partially charged atoms (positively charged: white with blue dashed outline; negatively charged: red with red dashed outline)

electrostatic interactions increases rapidly ($\mathcal{O}N_{pc}^2$) as the number of partially charged atoms increases.

4.8 | Effects not modeled

Attractive van der Waals forces and the effects of solvent are not modeled for simplicity. In the case of solvent, while solvent effects, in particular entropy and polarity, obviously make important contributions to protein structure, even a simple implicit solvent model such as the solvent accessible surface area (SASA)-based models commonly implemented in molecular dynamics software[25] would substantially increase the computational cost.

4.9 | Direct interaction

Prefab atom groups can be "grabbed" and manipulated directly using the motion controllers. Direct grabbing is the routine way that object manipulation is implemented in many VR games. The grabbed prefab unit is directly attached to the user's hand (controller), and thus inherits the hand position and rotation (transform), giving the user-intuitive 6DoF control.

Although direct interaction is highly intuitive and responsive, it has some inevitable issues. Grabbing and manipulating a prefab unit effectively take control of its transform away from the underlying physics governing the peptide behavior. Reconciling the grabbed object movements with those of the other connected prefab units that are controlled only by the underlying physics introduces effectively unlimited forces/torques. It is therefore possible to inadvertently distort bond lengths and angles. In addition, large rapid movements have the potential to generate "explosive" oscillations as the fixed time step simulated physics struggles to reconcile large instantaneous changes. It should be noted that this can also be a problem in more sophisticated molecular dynamics simulations, but may be rectified with the development of more sophisticated controllers. In the meantime, it can be argued that this is in fact a favorable feature with regard to engaging students that enhance the richness of the VR environment as well as initiating conversations about why it occurs and therefore a deeper understanding of the underlying physics.

4.10 | Remote interaction

Prefab atom groups can also be interacted with at a distance. Pointing a motion controller at an atom group highlights the group that can then be "tractor beamed" directly toward or away from the user. This functionality is further extended with a "remote grab" interaction, which allows the user to intuitively push, pull, and tangentially drag an atom group at a distance. This feature works by calculating an appropriate translational vector from the user's motion controller gesture and applying an impulse force to the remotely grabbed object. Although direct grabbing is perhaps more intuitive for a new user, the remote-interaction approach turns out to be very valuable for manipulating polypeptides.

4.11 | Dynamics

The forces resulting from the underlying physics are always heavily damped. Without this, the prefab's rigidbodies would be prone to acquiring large velocities particularly when being directly interacted with by the user. Damping is achieved by all rigidbodies having empirical drag factors enabled for translational and angular motions (Table S8). The overall scaling of these drag values can be adjusted by the user via a slider but can never be reduced to zero. Selected residues may be "frozen," which sets the drag parameters for the associated prefab units to infinity.

"Jiggle" dynamics, notionally equivalent to thermal motion, are achieved by applying random impulse forces to each of the rigidbodies. The impulse forces are

randomized every frame and the overall scale factor is empirical. These forces are applied to the center of mass of the rigidbodies, indicating that, as a consequence, prefab units (such as CH_3 groups) do not spin as much as is observed in molecular dynamics simulations.

ACKNOWLEDGMENTS

D.G.D., J.A.G., and J.R.A. acknowledge support from the University of Auckland Vice Chancellor's Strategic Development Fund. J.A.G. and D.G.D. also acknowledge support from the School of Chemical Sciences, University of Auckland. J.R.A. is additionally supported by a Rutherford Discovery Fellowship (15-MAU-001).

ORCID

Jane R. Allison https://orcid.org/0000-0002-5699-1726

REFERENCES

1. Trindade J, Fiolhais C, Almeida L. Science learning in virtual environments: A descriptive study. Brit J Educ Tech. 2002;33:471–488.
2. Mikropoulos TA, Natsis A. Educational virtual environments: A ten-year review of empirical research (1999–2009). Comput Educ. 2011;56:769–780.
3. Goddard TD, Brilliant AA, Skillman TL, et al. Molecular visualization on the holodeck. J Mol Biol. 2018;430:3982–3996.
4. Bennie S, Ranaghan K, Deeks H, et al. Teaching enzyme catalysis using interactive molecular dynamics in virtual reality. J Chem Educ. 2019; https://pubs.acs.org/doi/abs/10.1021/acs.jchemed.9b00181.
5. Johnston APR, Rae J, Ariotti N, et al. Journey to the centre of the cell: Virtual reality immersion into scientific data. Traffic. 2018;19:105–110.
6. Stefani C, Lacy-Hulbert A, Skillman T. ConfocalVR: Immersive visualization for confocal microscopy. J Mol Biol. 2018;430:4028–4035.
7. Stone JE, Kohlmeyer A, Vandivort KL, Schulten K. Immersive molecular visualization and interactive modeling with commodity hardware. In: Bebis G, Boyle R, Parvin B, et al., editors. Advances in visual computing. Berlin, Heidelberg: Springer, 2010; p. 382–393.
8. Norrby M, Grebner C, Eriksson J, Boström J. Molecular rift: Virtual reality for drug designers. J Chem Inf Model. 2015;55:2475–2484.
9. Grebner C, Norrby M, Enström J, et al. 3D-lab: A collaborative web-based platform for molecular modeling. Future Med Chem. 2016;8:1739–1752.
10. Salvadori A, Del Frate G, Pagliai M, Mancini G, Barone V. Immersive virtual reality in computational chemistry: Applications to the analysis of QM and MM data. Intl J Quant Chem. 2016;116:1731–1746.
11. Balo AR, Wang M, Ernst OP. Accessible virtual reality of biomolecular structural models using the Autodesk molecule viewer. Nat Methods. 2017;14:1122–1123.
12. Borrel A, Fourches D. RealityConvert: A tool for preparing 3D models of biochemical structures for augmented and virtual reality. Bioinformatics. 2017;33:3816–3818.
13. Zheng M, Waller MP. ChemPreview: An augmented reality-based molecular interface. J Mol Graph Model. 2017;73:18–23.
14. O'Connor M, Deeks HM, Dawn E, et al. Sampling molecular conformations and dynamics in a multiuser virtual reality framework. Sci Adv. 2018;4:eaat2731.
15. Ratamero EM, Bellini D, Dowson CG, Römer RA. Touching proteins with virtual bare hands. J Comput Aid Mol Des. 2018;32:703–709.
16. Kingsley LJ, Brunet V, Lelais G, et al. Development of a virtual reality platform for effective communication of structural data in drug discovery. J Mol Graph Model. 2019;89:234–241.
17. O'Connor MB, Bennie SJ, Deeks HM, et al. Interactive molecular dynamics in virtual reality from quantum chemistry to drug binding: An open-source multi-person framework. J Chem Phys. 2019;150:220901.
18. Schrodinger, LLC. The PyMOL molecular graphics system, Version 1.8. (2015).
19. Humphrey W, Dalke A, Schulten K. VMD: Visual molecular dynamics. J Mol Graph Model. 1996;14:33–38.
20. Kolasinski EM. Simulator sickness in virtual. Environments. 1995. https://apps.dtic.mil/docs/citations/ADA295861
21. Bondi A. van der Waals Volumes and Radii. J Phys Chem. 1964;68:441–451.
22. Shahbazi Z. Mechanical model of hydrogen bonds in protein molecules. Am J Mechan Engin. 2015;3:47–54.
23. Reif MM, Hünenberger PH, Oostenbrink C. New interaction parameters for charged amino acid side chains in the GROMOS force field. J Chem Theory Comput. 2012;8:3705–3723.
24. Reif MM, Winger M, Oostenbrink C. Testing of the GROMOS force-field parameter set 54A8: Structural properties of electrolyte solutions, lipid bilayers, and proteins. J Chem Theory Comput. 2013;9:1247–1264.
25. Fraternali F, van Gunsteren WF. An efficient mean solvation force model for use in molecular dynamics simulations of proteins in aqueous solution. J Mol Biol. 1996;256:939–948.

SUPPORTING INFORMATION

Additional supporting information may be found online in the Supporting Information section at the end of this article.

How to cite this article: Doak DG, Denyer GS, Gerrard JA, Mackay JP, Allison JR. Peppy: A virtual reality environment for exploring the principles of polypeptide structure. *Protein Science*. 2020;29:157–168. https://doi.org/10.1002/pro.3752