

# **Multiple Dynamic Pivots Rig for 3D Biped Character Designed along Human-Computer Interaction Principles**

by

Mihaela D. Petriu

A thesis submitted to the Faculty of Graduate and Postdoctoral Affairs in  
partial fulfillment of the requirements for the degree of

Master of Computer Science

in

Human-Computer Interaction

Carleton University  
Ottawa, Ontario

© 2020, Mihaela D. Petriu

## **Abstract**

This thesis is addressing the design, implementation and usability testing of a modular extensible animator-centric authoring tool-independent 3D Multiple Dynamic Pivots (MDP) biped character rig. An objective of the thesis is to show that the design of the character rig must be independent of its platform, cater to the needs of the animation workflow based on HCI principles (particularly that of Direct Manipulation) and be modular and extensible in order to build and grow as needed within the production pipeline. In the thesis, the proposed rig design is implemented in Maya, the most widely used and taught commercial 3D animation authoring tool. Another thesis objective is to perform usability testing of the MDP rig with animation students and professionals, in order to gauge the new design's intuitiveness for those relatively new to the trade and those already steeped in its decades of accumulated technical idiosyncrasies.

## **Acknowledgements**

My deepest gratitude goes to my supervisor Dr. Chris Joslin for his constant guidance and support.

I would also like to thank my family for their continuous help and encouragement.

Finally, I thank my friend Sandra E. Hobbs for her constructive chaos that shed light on my design's weaknesses.

# Table of Contents

<b>Abstract</b> .....	<b>ii</b>
<b>Acknowledgements</b> .....	<b>iii</b>
<b>Table of Contents</b> .....	<b>iv</b>
<b>List of Tables</b> .....	<b>vii</b>
<b>List of Figures</b> .....	<b>viii</b>
<b>List of Appendices</b> .....	<b>xvi</b>
<b>Chapter 1: Introduction</b> .....	<b>1</b>
1.1    Motivation and Objectives .....	1
1.2    Rationale and Benefits of Thesis Contributions .....	5
1.3    Thesis Contents .....	7
<b>Chapter 2: Background and State of the Art</b> .....	<b>8</b>
2.1    Principles of Animation.....	8
2.1.1    Squash & Stretch.....	10
2.1.2    Exaggeration .....	11
2.2    Animation Workflow.....	11
2.3    The Human-Computer Interaction Principles for Character Rigs .....	13
2.4    Animator-centric Rigs in the Animation Industry .....	14
2.4.1    Authoring Tool-Based Autorigs.....	15
2.4.2    Animator-Friendly Rigging.....	15
2.5    The State of the Biped Rig and Thesis Goals .....	18
2.5.1    Spine.....	19
2.5.2    Neck & Head.....	22
2.5.3    Legs & Arms .....	22

2.5.4	IK Feet & Hands .....	25
<b>Chapter 3: Multiple Dynamic Pivots Rig.....</b>		<b>29</b>
3.1	Rig Controller.....	35
3.2	Spine.....	35
3.3	Limbs.....	39
3.3.1	FK.....	39
3.3.2	MK .....	40
3.3.3	IK .....	42
3.3.4	DK.....	44
3.3.5	Roll.....	46
3.4	IK/DK Feet .....	47
3.5	IK/DK Hand .....	49
3.6	Sticky Finger .....	52
3.7	Follow/Free Connectors .....	55
<b>Chapter 4: MDP Rig Implementation.....</b>		<b>59</b>
4.1	Sub-Rig Switching.....	60
4.2	Word Alignment.....	60
4.3	Aligning IK.....	61
<b>Chapter 5: User Interface of the MDP Rig.....</b>		<b>62</b>
5.1	swithUI.....	63
5.1.1	Spine.....	63
5.1.2	Limb.....	66
5.1.3	Foot .....	69
5.1.4	Hand.....	71
5.1.5	Finger .....	74

5.1.6	Follow/Free .....	76
5.2	selKeyUI.....	79
5.2.1	Intro & Special .....	79
5.2.2	Selection.....	80
5.2.3	Keying.....	80
<b>Chapter 6: Usability Study.....</b>		<b>82</b>
6.1	Methodology.....	82
6.1.1	Survey .....	82
6.1.2	Participants.....	83
6.2	Result Discussion .....	84
<b>Chapter 7: Conclusions .....</b>		<b>91</b>
7.1	Discussion of Contributions .....	91
7.2	Limitations.....	91
7.3	Directions for Future Research.....	92
<b>Appendices.....</b>		<b>93</b>
	Appendix A Research Ethics Clearance.....	93
<b>References .....</b>		<b>96</b>

## List of Tables

Table 6.1	Results of the MDP rig usability test. ....	85
-----------	---	----

## List of Figures

Figure 2.1 IK wrist controller in Gimbal mode. ....	12
Figure 2.2 IK wrist controller animation with clean graph editor values. ....	12
Figure 2.3 IK wrist controller animation with messy graph editor values.....	13
Figure 2.4 Traditional rig (Animation Mentor's Stewart) arm blending from FK on keyframe 0 to IK on keyframe 4. The user interaction element of the transition animation is the "Fk Ik Blend" numerical field on the right.....	19
Figure 2.5 Character Studio Biped spine. ....	20
Figure 2.6 Animation Mentor Bishop FK spine rig.....	21
Figure 2.7 Animation Mentor Bishop IK spine rig.....	21
Figure 2.8 Character Studio Biped arm rig.....	23
Figure 2.9 Animation Mentor Bishop PV knee. ....	24
Figure 2.10 Bishop PV meets knee.....	25
Figure 2.11 Biped foot – red dot on the heel is pivot point. ....	26
Figure 2.12 Bishop foot lifting away from its main pivot. ....	27
Figure 2.13 Bishop digits and their controlling numerical fields. ....	28
Figure 3.1 The MDP IK Foot switching pivot modes from Ball (green) to Toe tip (pink) to Ankle (blue) to Heel (tan) and back to Ball in a walk animation. ....	30
Figure 3.2 The mode switching attributes of the MDP rig organized in a logical hierarchy under the rig's master controller.....	31
Figure 3.3 The motion trails of the left wrist and right ankle during a walk cycle – the blue trail is in the past and the pink in the future of the current frame. ....	32
Figure 3.4 Double IK leg translation controllers with translate widget active on knee....	33

Figure 3.5 Knee FK rotation controller with rotate widget active. ....	33
Figure 3.6 IK limb scaling controllers with scale widget active on knee. ....	34
Figure 3.7 Controllers of Spine1.....	37
Figure 3.8 Controllers of Spine2.....	37
Figure 3.9 Controllers of Spine3.....	37
Figure 3.10 Controllers of Spine4.....	37
Figure 3.11 Motion from Spine1 pivot. ....	38
Figure 3.12 Motion from Spine2 pivot – lower and upper hierarchies rooted at pivot move independently of each other. ....	38
Figure 3.13 Motion from Spine3 pivot – lower and upper hierarchies rooted at pivot move independently of each other. ....	38
Figure 3.14 Motion from Spine4 pivot. ....	38
Figure 3.15 Motion from Spine1 pivot with stretch (left) and squash (right).....	38
Figure 3.16 Controllers of the FK limbs.....	39
Figure 3.17 Motion of the FK arm.....	40
Figure 3.18 Motion of the FK leg with upper-bone stretch and lower-bone squash (left) and upper-bone squash and lower-bone stretch (right).....	40
Figure 3.19 Controllers of the MK limbs.....	41
Figure 3.20 Motion of the MK arm. ....	41
Figure 3.21 Motion of the MK leg with upper-bone stretch and lower-bone squash (left) and upper-bone squash and lower-bone stretch (right).....	41
Figure 3.22 Controllers of the IK limbs.....	43

Figure 3.23 The IK pole is the straight line between the top and end joints of the IK chain – the pole that the IK chain rotates about. ....	43
Figure 3.24 Motion of the IK arm.....	44
Figure 3.25 Motion of the IK leg with whole limb squash (left) and stretch (right). ....	44
Figure 3.26 Controllers of the DK limbs. ....	45
Figure 3.27 Motion of the DK leg with upper-bone squashed and lower-bone stretched (left) and upper-bone stretched and lower-bone squashed (right). ....	45
Figure 3.28 Motion of the DK arm with length of the bones resets at the extreme poses.	46
Figure 3.29 Candywrapping of the geometry mesh as shoulder, hip and ankle. ....	47
Figure 3.30 Candywrapping of the geometry mesh as shoulder, hip and ankle mitigated by the rotation of the limb roll controllers. ....	47
Figure 3.31 Controllers of the IK/DK Ankle pivot foot. ....	48
Figure 3.32 Controllers of the IK/DK Heel pivot foot.....	48
Figure 3.33 Controllers of the IK/DK Ball pivot foot. ....	48
Figure 3.34 Controllers of the IK/DK Toe tip pivot foot.....	48
Figure 3.35 Motion of the IK/DK Ankle pivot foot.....	49
Figure 3.36 Motion of the IK/DK Heel pivot foot.....	49
Figure 3.37 Motion of the IK/DK Ball pivot foot – the heel and toe lift controllers are independent from each other.....	49
Figure 3.38 Motion of the IK/DK Toe-tip pivot foot.....	49
Figure 3.39 Controllers of the IK/DK Wrist pivot hand. ....	50
Figure 3.40 Controllers of the IK/DK Finger1 – Thumb1 in this case – pivot hand. ....	50
Figure 3.41 Controllers of the IK/DK Finger2 – Thumb2 in this case – pivot hand. ....	50

Figure 3.42 Controllers of the IK/DK Finger3 – Thumb3 in this case – pivot hand. ....	50
Figure 3.43 Controllers of the IK/DK Finger4 – Thumb4 in this case – pivot hand. ....	50
Figure 3.44 Motion of the IK/DK Wrist pivot hand. ....	51
Figure 3.45 Motion of the IK/DK Finger1 – Index1 in this case – pivot hand.....	51
Figure 3.46 Motion of the IK/DK Finger2 – Index2 in this case – pivot hand.....	51
Figure 3.47 Motion of the IK/DK Finger3 – Index3 in this case – pivot hand.....	51
Figure 3.48 Motion of the IK/DK Finger4 – Index4 in this case – pivot hand.....	52
Figure 3.49 Controllers of the non-sticky Finger – Pinkie in this case – of the IK/DK Hand.....	52
Figure 3.50 Motion of the non-sticky Finger – Index in this case – of the IK/DK Hand.	53
Figure 3.51 Controllers of the Sticky1 Finger – Pinkie in this case – of the IK/DK Hand. ....	53
Figure 3.52 Controllers of the Sticky2 Finger – Pinkie in this case – of the IK/DK Hand. ....	53
Figure 3.53 Controllers of the Sticky3 Finger – Pinkie in this case – of the IK/DK Hand. ....	54
Figure 3.54 Motion of the Sticky1 Finger – Index Sticky1 in this case – of the IK/DK Hand.....	54
Figure 3.55 Motion of the Sticky2 Finger – Index Sticky2 in this case – of the IK/DK Hand.....	54
Figure 3.56 Motion of the Sticky3 Finger – Index Sticky3 in this case – of the IK/DK Hand.....	55
Figure 3.57 All the Follow-Parent rotation controllers of the whole body.....	55

Figure 3.58 All the Free-of-Parent rotation controllers of the whole body. ....	55
Figure 3.59 Motion of the Follow-Parent Head.....	56
Figure 3.60 Motion of the Free-of-Parent Head. ....	56
Figure 3.61 Motion of the Follow-Parent Spine-Neck connector.....	56
Figure 3.62 Motion of the Free-of-Parent Spine-Neck connector. ....	56
Figure 3.63 Motion of the Follow-Parent Spine-Pelvis connector. ....	56
Figure 3.64 Motion of the Free-of-Parent Spine-Pelvis connector.....	56
Figure 3.65 Motion of the Follow-Parent Arm connector. ....	57
Figure 3.66 Motion of the Free-of-Parent Arm connector.....	57
Figure 3.67 Motion of the Follow-Parent Leg connector. ....	57
Figure 3.68 Motion of the Free-of-Parent Leg connector.....	57
Figure 3.69 Motion of the Follow-Parent FK/MK Wrist connector.....	57
Figure 3.70 Motion of the Free-of-Parent FK/MK Wrist connector. ....	57
Figure 3.71 Motion of the Follow-Parent FK/MK Ankle connector.....	58
Figure 3.72 Motion of the Free-of-Parent FK/MK Ankle connector.....	58
Figure 5.1 The switchUI legend.....	63
Figure 5.2 Spine1 button triggers mode switch to Spine1 controllers to create Spine1 pivot-centric motion.....	64
Figure 5.3 Spine2 button triggers mode switch to Spine2 controllers to create Spine2 pivot-centric motion.....	64
Figure 5.4 Spine3 button triggers mode switch to Spine3 controllers to create Spine3 pivot-centric motion.....	65

Figure 5.5 fig. Spine4 button triggers mode switch to Spine4 controllers to create Spine4 pivot-centric motion.....	65
Figure 5.6 Arm/Leg IK button triggers mode switch to Arm/Leg IK controllers to create Arm/Leg IK motion. ....	66
Figure 5.7 Arm/Leg MK button triggers mode switch to Arm/Leg MK controllers to create Arm/Leg MK motion.....	67
Figure 5.8 Arm/Leg IK button triggers mode switch to Arm/Leg IK controllers to create Arm/Leg IK motion. ....	68
Figure 5.9 Arm/Leg DK button triggers mode switch to Arm/Leg DK controllers to create Arm/Leg DK motion.....	69
Figure 5.10 Foot Ankle IK/DK button triggers mode switch to Foot Ankle IK/DK controllers to create Foot Ankle-centric motion. ....	70
Figure 5.11 Foot Heel IK/DK button triggers mode switch to Foot Heel IK/DK controllers to create Foot Heel-centric motion. ....	70
Figure 5.12 Foot Ball IK/DK button triggers mode switch to Foot Ball IK/DK controllers to create Foot Ball-centric motion. ....	71
Figure 5.13 Foot Toe-tip IK/DK button triggers mode switch to Foot Toe-tip IK/DK controllers to create Foot Toe tip-centric motion. ....	71
Figure 5.14 Hand Wrist IK/DK button triggers mode switch to Hand Wrist IK/DK controllers to create Hand Wrist-centric motion.....	72
Figure 5.15 Hand Finger1 IK/DK button triggers mode switch to Hand Finger1 IK/DK controllers to create Hand Finger1-centric motion. ....	72

Figure 5.16 Hand Finger2 IK/DK button triggers mode switch to Hand Finger2 IK/DK controllers to create Hand Finger2-centric motion. ....	73
Figure 5.17 Hand Finger3 IK/DK button triggers mode switch to Hand Finger3 IK/DK controllers to create Hand Finger3-centric motion. ....	73
Figure 5.18 Hand Finger4 IK/DK button triggers mode switch to Hand Finger4 IK/DK controllers to create Hand Finger4-centric motion. ....	74
Figure 5.19 Hand IK/DK Finger Sticky1 button triggers mode switch to Finger Sticky1 controllers to create Finger Sticky1 motion. ....	75
Figure 5.20 Hand IK/DK Finger Sticky2 button triggers mode switch to Finger Sticky2 controllers to create Finger Sticky2 motion. ....	75
Figure 5.21 Hand IK/DK Finger Sticky3 button triggers mode switch to Finger Sticky3 controllers to create Finger Sticky3 motion. ....	76
Figure 5.22 Hand IK/DK Finger Sticky Off button triggers mode switch to default controllers to create Finger Sticky1 motion. ....	76
Figure 5.23 Follow/Free Rotation toggled triggers mode switch to Free-of-Parent Rotate controller if in Follow mode to create motion free of the rotation of the parent controller. ....	77
Figure 5.24 Follow/Free Rotation toggled triggers mode switch to Free-of-Parent Rotate controller if in Follow mode to create motion free of the rotation of the parent controller. ....	77
Figure 5.25 Follow/Free Rotation toggled triggers mode switch to Follow-Parent Rotate controller if in Free mode to create motion following the rotation of the parent controller. ....	78

Figure 5.26 Follow/Free Rotation toggled triggers mode switch to Free-of-Parent Rotate controller if in Follow mode to create motion free of the rotation of the parent controller. .....	78
Figure 5.27 The selection/key UI (selKeyUI) legend. ....	79
Figure 6.1 Spine opinion level distribution. ....	86
Figure 6.2 Spine experience level distribution. ....	86
Figure 6.3 Limbs opinion level distribution. ....	86
Figure 6.4 Limbs experience level distribution. ....	86
Figure 6.5 Foot opinion level distribution. ....	86
Figure 6.6 Foot experience level distribution. ....	86
Figure 6.7 Hand opinion level distribution. ....	87
Figure 6.8 Hand experience level distribution. ....	87
Figure 6.9 Follow/Free opinion level distribution. ....	87
Figure 6.10 Follow/Free experience level distribution. ....	87
Figure 6.11 Overall MDP rig opinion level distribution.....	87
Figure 6.12 Overall rig experience level distribution. ....	87

## List of Appendices

Appendix A Research Ethics Clearance .....	93
--	----

## **Chapter 1: Introduction**

### **1.1 Motivation and Objectives**

Conceptually 3D character animation is the creation of a performance through the medium of a virtual actor. In practice, 3D character animation is the manipulation of the position, rotation and scale of dozens of objects, called “controllers” because they control the body parts of the virtual actor, and the recording of the resultant position, rotation and scale values on particular frames of the timeline of the performance. The controllers are organized in sets of hierarchy – the ankle controller is a child of the knee controller and knee controller is a child of the hip controller, for example – and all the sets in concert are used to pose the virtual actor, also known as the “character”. Poses that are key to the motion of the performance or, more specifically, the “animation”, are recorded on “keyframes” – these are the poses that define the extremes of the motions of the controllers.

There is potentially a multitude of frames per second of animation with 24 and 30 frames per second being the most common – much animating effort goes into creating a few seconds of animation. Animation is an iterative process of trial-and-not-quite-right as body part poses, the frame on which those poses happen, and the rate at which the transitions between poses happen are all refined until the action conveys the intended physicality and emotion. The believability of the animation is the most important, and difficult to achieve, in humanoid character animation since the audience feels instinctually how such a character is supposed to move since they have spent a significant portion of their time watching and interacting with people since childhood. Any deviation from the believable can topple the character into the Uncanny Valley (Mori, 2012) of

being initially accepted as a person – even if stylized – but then not moving in a way that’s believable enough causing the audience to reject the character with greater force than if they hadn’t accepted it as a person initially. This danger of being worse due to not being quite good enough makes the need for exact animation controls of 3D characters even more necessary.

The perils of the Uncanny Valley do not haunt 2D animation as it does 3D, and 2D animators do have far more control of their character from frame-to-frame than 3D animators do. This is due to 2D simply not having the realism that 3D does – 3D has lighting, shadows, textures and hair that the audience believes could physically exist while 2D has lines on screen that look like lines on a screen. 2D animation never gets close enough up the believability mountain to risk falling into the Uncanny Valley. 2D animation is the progenitor of the animation industry and the core concepts of animation were formalized during its ascendance to popularity (Thomas et al, 1981) and while 3D animation has inherited its predecessor’s concepts, its practice is distinctly different. Animating in 3D is like puppetry rather than drawing. All the things that 2D animators must worry about because they have to create the illusion of a third dimension in only two, such as perspective and depth, 3D animators get for free at the expense of using a more complicated tool than pencil and paper.

3D animators use specialized software on computers and the tool they animate with is called a “rig” which is the collection of all those hierarchically organized controllers the animators manipulate within the software. These rigs are designed by the “Technical Directors” and built by the “riggers” within a 3D animation production house with

varying degrees of input from the “Directors of Animation” and the animators themselves.

The purpose of 3D animation rigs naturally lends itself to the Human-Computer Interaction (HCI) principle known as “Direct Manipulation” (Shneiderman, 1983) (Hutchins et al., 1986) (Shneiderman, 1988) (Shneiderman, 1997) (Ziegler et al., 1988) (Frohlich, 1993) which requires:

1. constant display of the object of the interaction
2. interaction that is physical, rather than abstract, in nature
3. constant feedback of the interaction that includes the ability to reverse the outcome of the interaction and achieve the outcome incrementally

The task of 3D animation requires all three criteria of Direct Manipulation:

1. animators need to constantly see the controllers that they are manipulating and the character that is being controlled
2. animators physically move their mouse to position, rotate, and scale the controllers within the 3D scene to create the illusion of a performance
3. animators see the outcome of their controller manipulations as they do them, are able to undo them instantly and incrementally achieve the desired pose and timing through incremental changes

However, the rigs that 3D animators have traditionally used have not fully leveraged the natural fit of Direct Manipulation to the design of 3D character rigs.

The objectives of this thesis are the design, implementation, and usability testing of the Multiple Dynamic Pivots (MDP) 3D biped character rig. The defining features of the MDP rigs design are:

- Animator-centricity – the rig design is to be top-down starting with the requirements of 3D animators to give them the precise control over the motion of the characters while freeing them from the technical tasks that can be automated
- 3D authoring tool-independence – the design should be implementable in any 3D authoring software such as Maya, 3dmax, Cinema4D, or Blender
- Modularity – the MDP rig is to be collective of simple specialized sub-rigs that are independent of each other so that individual sub-rigs can be modified without affecting the rest of the rig
- Extensibility – sub-rigs can be added to the MDP rig as needed without any effect on the existing sub-rigs

Keyframe 3D character animation is a high-investment occupation requiring considerable amounts of time, skill and talent from its practitioners. However, twenty-five years after the theatrical release of Pixar's "Toy Story" (Pixar, 1995), whose artistic 3D character animation methods have spread and are now considered classic, the workflow of professional animators remains largely unstudied and unoptimized by the current standards of HCI. In particular, 3D character animation is conspicuously divorced from Shneiderman's concept of Direct Manipulation to which such animation – the manipulation of 3D objects in virtual time and space – should be a natural fit. To be effective, the optimization of the character animator's workflow requires an understanding of both the principles of animation and those of HCI.

The motivation of the thesis is to analyze the mechanics of 3D character animation and design a rig to meet the needs of 3D character animators using HCI principles, such as Direct Manipulation, as the guidelines of the design process.

## **1.2 Rationale and Benefits of Thesis Contributions**

The *contributions to knowledge of the thesis* represent the analysis of the fundamental needs of 3D animators and the application of HCI principles to meet those needs. Most existing research in rigging is focused on the technicalities of implementation rather than design (Bhati et al., 2015). Most existing research into animation is about addressing the needs of 2D animators into making the transition to 3D (Mao et al., 2007), exploring tangential display technologies such as Virtual Reality (Bowman et al., 2001) (Zhou, 2017) and Augmented Reality (Bowman et al., 2005) (Ye et al., 2020), tangential input technologies such as touch- and stylus-controls (Walther, 2014) or, more recently, into bypassing the animators altogether in favour of Artificial Intelligence (AI), as reflected by the 2020 crop of SIGGRAPH'20 papers on character animation, which are about using motion-capture, AI and Machine Learning to produce animation while having as little human input as possible (Aberman et al., 2020a) (Aberman et al., 2020b) (Eom et al., 2020) (Harvey et al., 2020) (Holden et al., 2020) (Kwon et al., 2020) (Ling et al., 2020) (Luo et al., 2020) (Merel et al., 2020) (Starke et al., 2020).

Even the Walt Disney Animation Studios – the formalizers of the Principles of Animation – have published very few papers on 3D animation per se. According to the list of Walt Disney Animation Studios publications and talks between 1996-2019 found at (Walt Disney, 2019), only the following three papers/talks are on animation: i) one paper about the workflow of 2D animators (Whited et al., 2010); ii) one SIGGRAPH 2012 talk about

making rigging code accessible to animators (Smith et al., 2012) and iii) one SIGGRAPH 2015 talk acknowledging 3D animators' need for realtime playback, but concerned only with the technical aspects of providing it (Lin et al., 2015). The majority of recent Walt Disney Animation Studios publications and talks listed at (Walt Disney, 2019), have been concerned with rendering and simulation.

As 3D rigging – the creation of a virtual puppet to be animated – is a discipline at the nexus of Computer Science, Human-Computer Interaction and Animation, it should be further explored and studied on those terms.

The *practical contributions* of the thesis – the design, implementation and usability testing of the MDP rig – aim to reduce the friction between the animators' imagination and their performance-by-proxy. Much of a character animator's time is consumed by the technical minutiae of using a particular 3D application, yet no character animator views their occupation as being the user of a software, a UI or even a rig. Instead, their art is to imbue characters with the illusion of life. Designed from the ground-up on the fundamentals of HCI and 3D animation, the resulting rig would be the instrument that facilitates the creative work of animators rather than impeding it. This would make for a more efficient and artistic workflow and product, as well as a more painless one, as less use of mouse and keyboard will mean fewer repetitive stress injuries. More intuitive rigs would also lower the barrier-to-entry for new animators – those who are interested in animation but frustrated by the technical minutiae could become animators without that impediment. This would result in more animators and more talent applied to the field of animation.

The animation industry would not only benefit from more and better animators, but also from Technical Directors and riggers that better understand the needs of animators and how

to design rigs to meet those needs. This would have the added benefit of reducing the acrimony between animation and rigging departments that can currently flare-up within the industry.

### **1.3 Thesis Contents**

The next chapter of the thesis, "Background & State of the Art", discusses the Principles of Animation formalized by Walt Disney, summarizes the workflow of 3D character animators, introduces the application-based autorigs that were designed specifically for character animation then details the state of each part of the character rig, its issues and how the thesis proposes to address those issues.

The third chapter, "Multiple Dynamic Pivots Rig", goes into details of how each part of the MDP rig is designed and the justification for that design.

The fourth chapter, "Implementation of the Multiple Dynamic Pivots Rig", expands upon the 3D techniques underpinning of the MDP rig.

The fourth chapter, "Multiple Dynamic Pivots Rig UI", describes how the UI button panels affect the MDP Rig and the justification for their design.

The fifth chapter, "Usability Study", describes the methodology of the MDP Rig usability study and discusses its results.

The sixth chapter, "Conclusions", summarizes the contributions of the thesis, the limitations of the work done so far and the directions for future research.

## Chapter 2: Background and State of the Art

This chapter discusses the Principles of Animation formalized by Walt Disney, summarizes the workflow of 3D character animators, describes the HCI principles best-suited to 3D animation, discusses the animator-centric rigs produced by the animation industry then details the state of each part of the character rig, its existing issues and how it goes against the relevant HCI principles.

### 2.1 Principles of Animation

The principles of animation were codified by Walt Disney (Thomas et al, 1981) and their continued importance to the then nascent field of 3D computer graphics were vouched for by John Lassiter, the founder of Pixar, in his 1987 paper "Principles of Traditional Animation Applied to 3D Computer Animation" (Lassiter, 1987). The Principles of Animation are as follows (those relevant to rigging are shown in bold):

1. **Squash & Stretch** - the deformation of the animated object in motion – the softer it's meant to look, the greater the deformation and the more rigid, the lesser the deformation.
2. Timing - the spacing of the object between animation frames – heavier/larger/sleepier/ depressed bodies have small spacing and lighter/smaller/energetic/exuberant bodies have large spacing.
3. Anticipation - the wind-up for an action, usually in the opposite direction of that of the action, like the drawing back of the arm before the fist is propelled forward in a punch.
4. **Staging** - the presentation of the action to the camera: this encompasses the placement of the characters and props and the paths of their motion in relation to

the "inanimate" background to make the action as "readable" as possible for the audience.

5. Follow Through and Overlapping Action – the other end of Anticipation: the "settling" of the animated object after the action leading into the wind-up for the next action.
6. Straight Ahead Action vs Pose-to-Pose Action – the two possible animation methods: Straight Ahead Action is animated from start to finish and tends to be spontaneous in its execution so it is usually used for frenetic action that is not meant to communicate much more than "Action!" while Pose-to-Pose is meticulously planned out from pose to pose before the in-between animation is done and is meant to hit very specific "beats" in the performance of the action especially if external cues like audio are involved.
7. Slow In and Out – the spacing of the animated object between frames is close together (slow) at the start of the action, becomes more spaced out (fast) in the middle of the action then becomes close again (slow) at the ending of the action. The rate at which this occurs is used by the animator to communicate mass, velocity and emotion.
8. **Arcs** – the path of the motion through the animation "space" usually describes an arc. Living beings, especially, rarely move themselves, or parts of themselves, in perfectly straight lines, so arcs come across as "organic" to the audience and straight lines as "robotic".
9. **Exaggeration** – the emphasis of the core idea of the animation such as the molasses movements of a constantly moping donkey that sighs "depressed" in every frame

or the rubber ball ballistics of an incessantly excited tiger that trills "exuberant" in every frame. The audience is left in no doubt as to the defining character moods of the donkey and the tiger.

10. **Secondary Action** – the motion of another object that is activated by the action of the primary object such as the flop of a dog's ears when she moves her head. The focus is the head movement and the ear flop the secondary action that make her seem more alive to the audience.
11. Appeal – aesthetic of characters, objects and setting and their motion that induces the audience to watch them – the "charisma" of animated productions.

### **2.1.1 Squash & Stretch**

Squash & Stretch, Staging, Arcs and Secondary Action are all principles that should be mechanically built into the character rig. Animators need to have control over the squashing and stretching of at least the spine, neck and limbs and, in extreme cases, the head, feet, hands, toes and fingers as well. The character rig as a whole, with all of its animation intact, needs to be able to be translated and rotated about the set, should the staging be changed after animation has started. Thus, the rig needs to give the animator clear control of what space – world, character, torso, limb, etc. – the animations are happening in. Arcs are intrinsic in the rotation of limbs in Forward Kinematics (FK) animation but can be more difficult to achieve in Inverse Kinematics (IK) translations. Therefore, rigs need to be built to make natural-looking arcing motions as achievable as possible in both FK and IK. Finally, the head, neck, shoulders, hands, hips and feet need to be able to rotate dependently, as well as independently, of their parent body parts to facilitate the animating of the drag and to settle their Secondary Action.

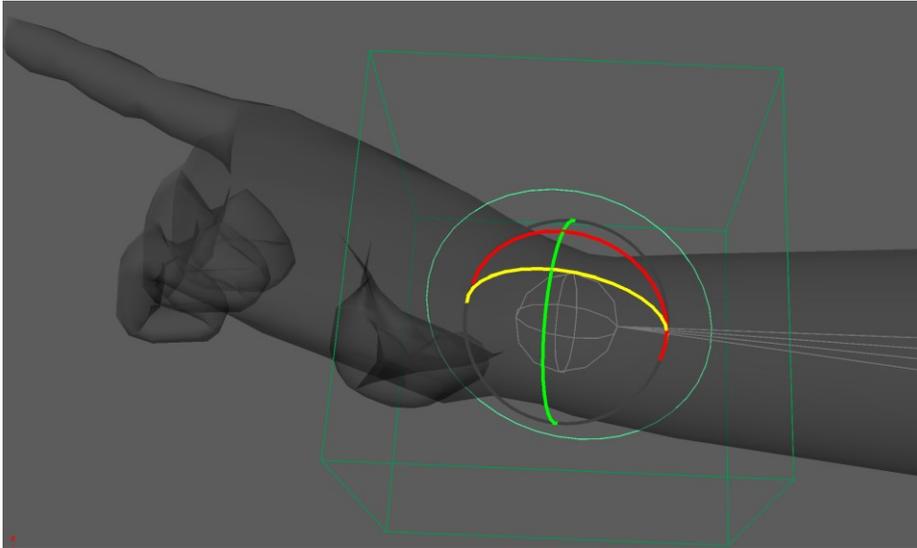
### **2.1.2 Exaggeration**

The principle of Exaggeration has an increasingly profound influence on the design of the character to rig, depending on how stylized or cartoonish the animation is intended to be. As discussed in Blair's 1994 book "Cartoon Animation" (Preston, 1994), the line of action of a character – the line tracing through the character's form between the longest extremes of an action – must clearly communicate to the audience the energy and intent of the action. Functionally, this means that the rigs for such cartoonish characters must have limbs and spines that are bendable to beyond the realistic. The capability of curving the character's line of action into C- and even S-shapes is a crucial requirement of these rigs, otherwise, animators will break the rig to achieve the extreme poses they require.

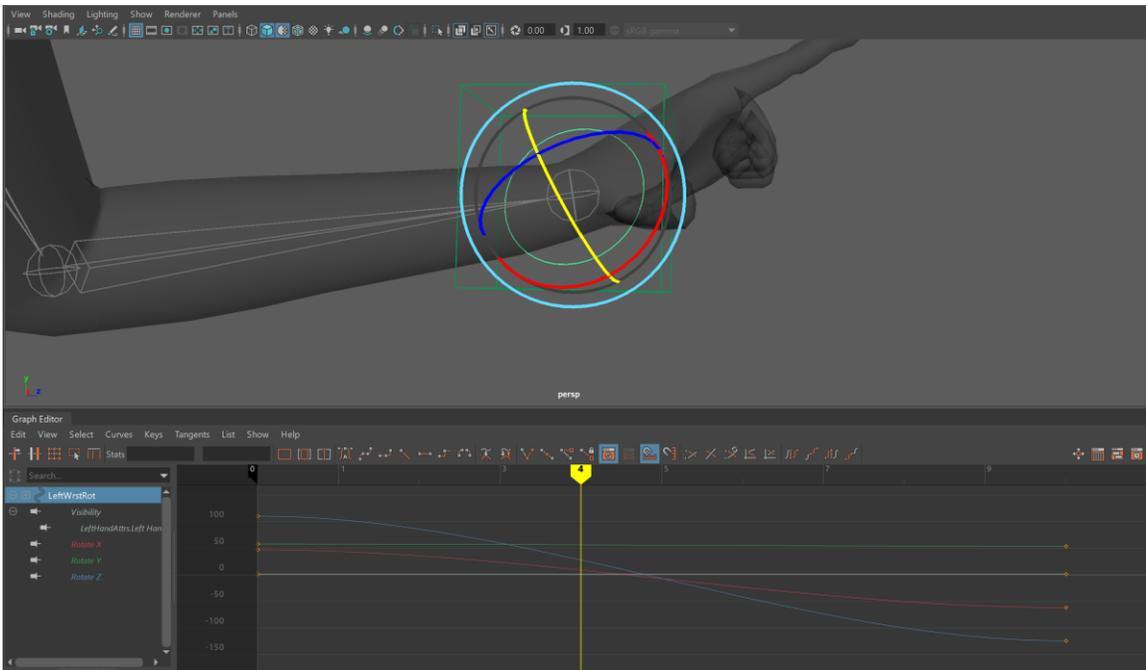
### **2.2 Animation Workflow**

The precise details of the animation workflow vary from artist to artist, but the one constant is that it is done in at least 2 stages: blocking and refinement. The *blocking stage* is the posing and keying of the character in the extremes of the actions they will perform at the estimated times they will perform them. It is vital at this stage to make it as frictionless as possible for the animator to produce clean and logical controller transformation values in the animation graph. The rotation of joints such as the shoulders and hips that have free rotation on all three axes can lead to sub-optimal software solutions between two extreme poses, leading the computer interpolated in-between animation to wild and unexpected contortions. Thus, the rig must be designed with a well thought-out axes rotation order and allow the animator to either work in Gimbal mode as shown in Figure 2.1 – showing them if they are about to create a "gimbal locked" (Popa, 1998) rotation where the rotation axes will cross – or facilitating the Local mode zero-ing out of transform values so that each

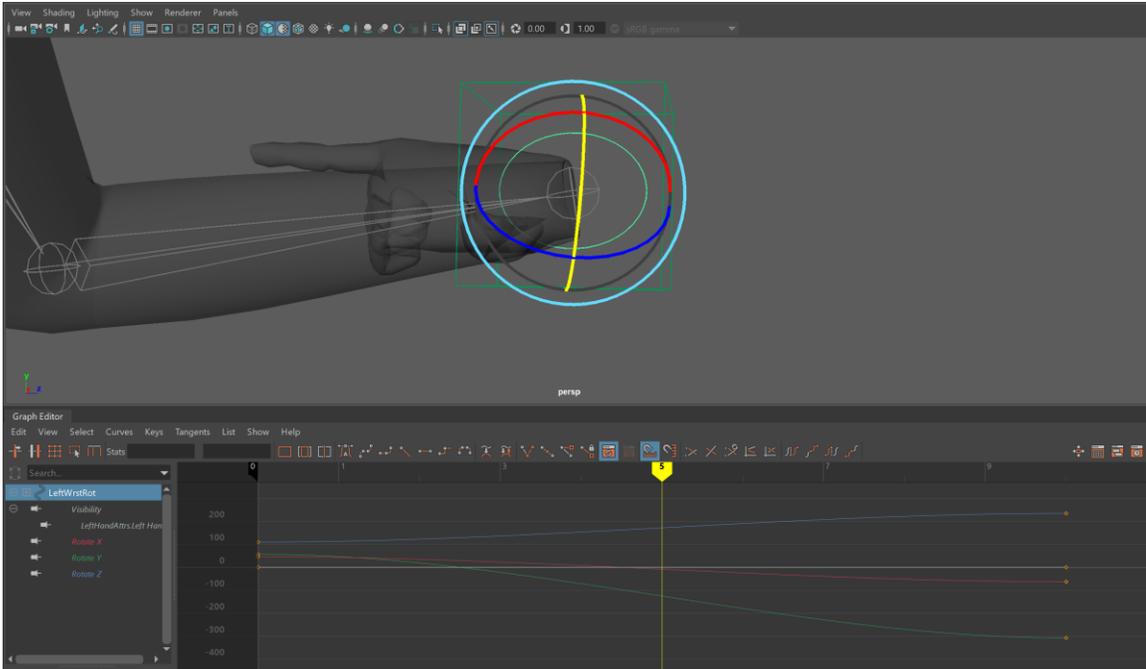
controller can be returned to its clean initial state before being posed into an extreme – leading to clean graph values and predictable interpolations. See Figure 2.2 and Figure 2.3 for clean and messy graph editor values, respectively.



**Figure 2.1 IK wrist controller in Gimbal mode.**



**Figure 2.2 IK wrist controller animation with clean graph editor values.**



**Figure 2.3 IK wrist controller animation with messy graph editor values**

The *refinement stage* is the most time-consuming and iterative of stages, as finer and finer details are tweaked, but it can also lead to major change of poses that can become particularly dire if overly burdened hierarchies exist in the rig. For instance, if a hierarchical controller structure was used to make the manipulation on different pivots of the foot or hand possible, and it was determined that the position of rotation of one of the "parent" pivots needed to be changed but the positions and rotations of its "children" pivots were to remain as they were, the animator would have a long process of counter-animation ahead of them, as they would have to animate the change in the parent while countering the non-change in the children. A well-designed rig needs to avoid such irreparable dependencies between its controllers.

### **2.3 The Human-Computer Interaction Principles for Character Rigs**

Shneiderman's Direct Manipulation (Shneiderman, 1983) (Hutchins et al., 1986)

(Shneiderman, 1988) (Shneiderman, 1997) (Ziegler et al., 1988) (Frohlich, 1993) is the

HCI principle best-suited to guide the design of 3D rigs based on its requirements for constant display of the interaction elements, physical interaction and realtime feedback, but more specific principles are needed for the design of the controllers of the rig. The most applicable are two of Norman's Design Principles (Norman, 2013):

- Affordances – the look, placement, feel, etc. of the object gives a clue as to how to interact with it, so controllers should be shaped and displayed in a way that gives the animator a clue to what kind of manipulation of the character they will perform.
- Mapping – the placement of interaction elements in relation to their effect, therefore the controllers should most logically be centred on the character joint that they are going to move, rotate or scale.

These principles will lead to multiple controllers centred on one joint that will need to be selected and moved with the mouse pointer. These will cause an accuracy versus speed trade-off as described by Fitts's Law (Fitts, 1954) (MacKenzie, 1992) which states that the further away the target element from the current position of the pointer and the smaller the element is, the longer it will take to accurately reach it. Thus the “grab-ability” of the controllers also dictates their shape and their placement and under which circumstances they are displayed.

#### **2.4 Animator-centric Rigs in the Animation Industry**

The animation industry has attempted to make rigs that cater to the requirements of character animators both within 3D authoring tools and by informal industrial experience-aggregate design guidelines. The authoring tool-based solutions are pre-made automatic rigs, called “autorigs”, that can be fitted to most two-legged, two-armed, one-headed

characters. The most formalized industry character rig design guidelines are from a series of instructional videos called “Animation-Friendly Rigging” (Schleifer, 2006).

#### **2.4.1 Authoring Tool-Based Autorigs**

There are professional-grade character autorigs available within Autodesk’s MotionBuilder, Maya and 3dsmax, but they are tellingly seldom used in productions doing keyframe animation. MotionBuilder’s and Maya’s HumanIK autorigs (Autodesk, 2019) are generally used for working with motion capture animation. The Character Studio Biped autorig in 3dsmax (Autodesk, 2018), however, is one of the most animator-centric autorigs designed commercially and possesses the greatly desirable features of bones that are simultaneously FK and IK in nature and animatable pivot shifting on the hands and feet. Nevertheless, it is inextricably dependent on 3dsmax and cannot be used in any other 3D authoring tool – this hampers its usability since not all animators will consider 3dsmax the most intuitive 3D authoring tool and few production pipelines would be willing to accommodate the authoring tool preference of every individual animator.

#### **2.4.2 Animator-Friendly Rigging**

Most production autorigs are designed, implemented, and maintained on a per-studio basis by an on-site team of Technical Directors and riggers who can agilely respond to the specific needs of their animation team. Development of those rigs is usually done under a Nondisclosure Agreement. However, the rigging professionals themselves still have had the opportunity to hone their craft over multiple productions in different studios and have gone on to write books, produce instructional videos and create instructional programs such as Rigging Dojo (Carey, 2020) and the CGSociety Workshops (CG Society, 2020a). Thus, character rigging conventions have emerged and a consensus "traditional character rig" can

be drawn from sources often referred to by the rigging community: "The Art of Rigging Volume 1" by Ritchie et al. (Ritchie et al., 2005), "Rig it Right!: Maya Animation Rigging Concepts" by O'Hailey (O'Hailey, 2013), and the "Animator Friendly Rigging" (AFR) SIGGRAPH 2006 Master Class by Jason Schleifer (Schleifer, 2006).

Schleifer's AFR design is particularly well-regarded, coming as it does from the Animation Lead on the original "Lord of the Rings" trilogy (NLC, 2001). The requirements of AFR are as follows:

1. Consistent/Predictable behaviour – the manipulation of rig controls must produce the results animators expect.
2. Simplified control structure - controller must be organized in a logical, intuitive manner with finer controls hidden until they are needed, lest they overwhelm the animators.
3. Easy to use - rigs must be as intuitive as possible to the animators and not burden the artistic process with onerous technical details.
4. Fast interaction - the time gap between manipulation and outcome should be as small as possible.
5. Minimal counter-animation - animators should have to do little, preferable no, "technical" animation to give illusion of one animation not causing another motion when in fact, it does.
6. Solve the problem - give the animators the tools to produce the motion needed, and not more or less.

7. Allows the animator to do their job without compromising their ideas - conceived performances of animated objects should not be comprised by the animator being unable to reproduce those performances on the provided rigs.

Tellingly, the most successful rig designs come from animators and not riggers – starting from a purely rigger perspective can lead to a discordant disconnect between the technical sensibilities of Technical Directors and the artistic sense of the Directors of Animation. This artistic-technical tension reflects the core struggle of rig design – finding the "just right" compromise between animator autonomy and animation automation.

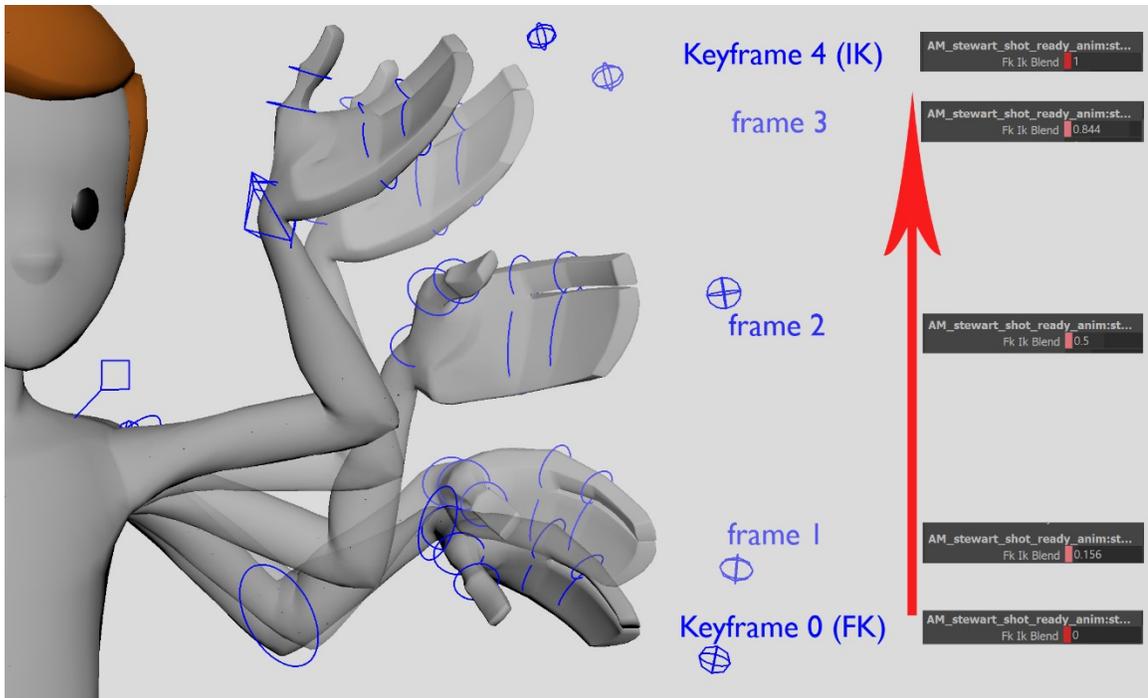
The issue remains that 3D animation is still such a young form of the art – the first major 3D animation feature film was released in 1995 and “The Lord of the Rings” that gave the AFR design such clout was released 6 years later. The animators that were so influential in the designing of the “traditional” 3D character rig used by the industry were themselves influenced by the concept of animation formed by 2D animation. That traditional character rig that is still used to teach and produce 3D animation on an industrial scale is still wedded to the traditional 2D animation idea of blending from pose-to-pose. Every pose is conceptualized as a discrete keyframe, and the animator must then wrangle the computer into acting as a lower pay-grade “inbetweenner” (Williams, 2002) to produce the frames of animation in between the keyframes. However, from a mechanical point of view, 3D characters have more in common with marionettes than lines on paper, but with each part of the character having multiple control rigs rather than just the one. The legs, for instance, can be animated by rotating the hip, knee and ankle joints in a Forward Kinematics (FK) fashion or by moving the ankle and having the rest of the leg joints rotate to follow the ankle in an Inverse Kinematics (IK) fashion. Therefore, a 3D character animator is actually

a puppeteer with the time control of a 2D animator, but the traditional rig design doesn't fully leverage that distinction or the full automation powers of the computers the animators use.

## **2.5 The State of the Biped Rig and Thesis Goals**

Character rigs are designed to be animated in the planned behaviour of the character in question and no more, as to not waste production resources on what will not be shown. For instance, an adult human character will probably not be required to roll around on the floor like a dog, unless that is explicitly called for in the pre-production storyboards. So specific body parts are designed to do what humanoid characters usually do: standing, walking and sitting and the transition between those actions.

Complications arise when more than those common actions are required, and the rigs become complicated multiplexes of rigs, which the animators must somehow blend between to achieve the desired performance. This blending is time-consuming and counter the Direct Manipulation (Shneiderman, 1983) (Hutchins et al., 1986) (Shneiderman, 1988) (Shneiderman, 1997) (Ziegler et al., 1988) (Frohlich, 1993) principles that controller interactions should be physical. In the case of manually blending between two rigs, the controllers themselves maybe be directly manipulable, but the blending itself is via an abstract number, as shown in Figure 2.4, that divorces the animator from directly manipulating the character physically.



**Figure 2.4 Traditional rig (Animation Mentor's Stewart) arm blending from FK on keyframe 0 to IK on keyframe 4. The user interaction element of the transition animation is the "Fk Ik Blend" numerical field on the right.**

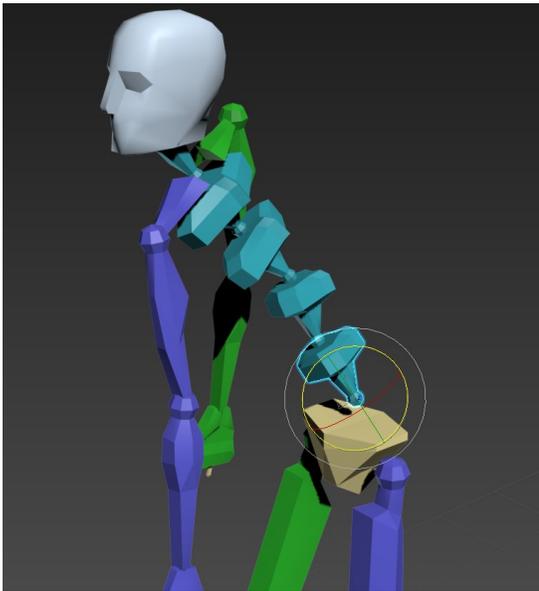
### 2.5.1 Spine

The spine is usually only required to achieve twisting, lateral swaying and back and forth bending between the pelvis and shoulders with the whole spine pivot being located at the pelvis. However, if the character is required to walk on their hands, fall on their back, swim or fly, this control structure become sub-optimal for the animator.

The Character Studio Biped rig as shown in Figure 2.5 is optimized for the twisting, swaying, and bending, but is capable of little else without a lot of work on the part of the animator. Since the Biped rig is essentially a blackbox, adding control structures to it on the rigging side for the animators is even more difficult.

Superficially, the Character Studio Biped spine seems to adhere to HCI principles – the controllers for directly manipulating the spine, look like they control a spine and are placed

along the segments of the spine. However, should all the spine segments need to be rotated from any point along the spine that's not the point where spine meets pelvis, the control fails since only the pelvis to lower spine to upper spine to neck controller hierarchy is present.



**Figure 2.5 Character Studio Biped spine.**

The Animation Mentor “Bishop” rig, as shown in Figure 2.6 and Figure 2.7, is more versatile. It has an FK sub-rig that can handle all the normal twisting, swaying and bending as well as an IK sub-rig that can be translated from the pelvis and the shoulders. However, while they can contort the character’s spine into a wide range of exaggerated poses, the controls can all-too-easily be separated from the character, which is unintuitive and confusing for the animators and make it difficult to avoid unnatural, and occasionally undesirable undulations of the spine that more resemble a snake than a human.

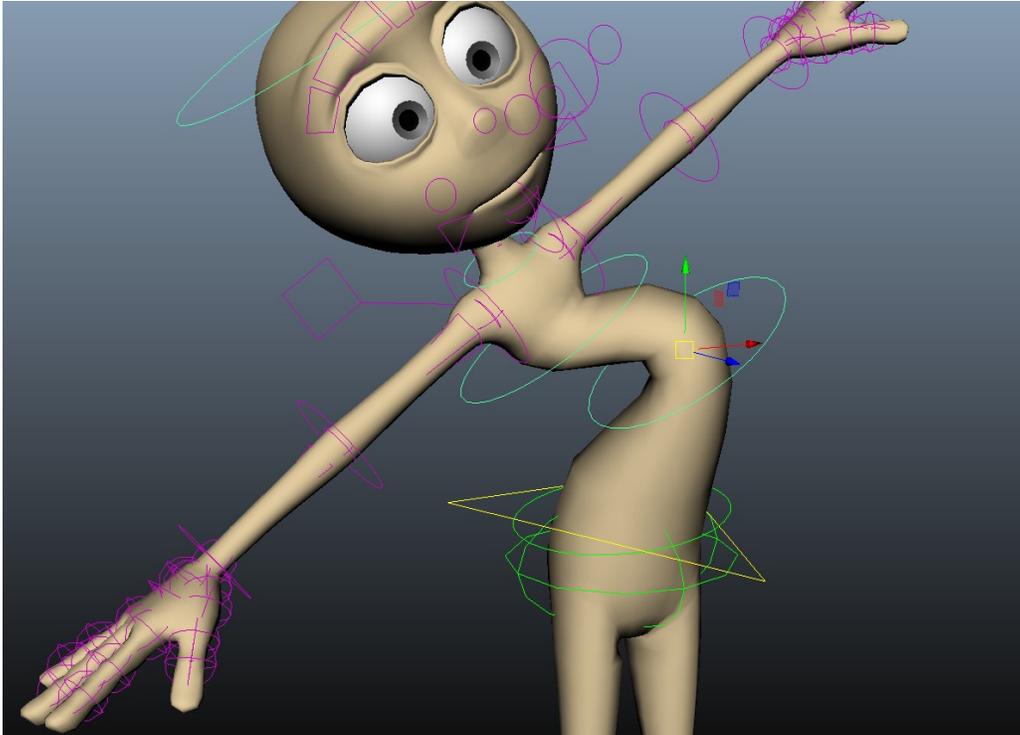


Figure 2.6 Animation Mentor Bishop FK spine rig.

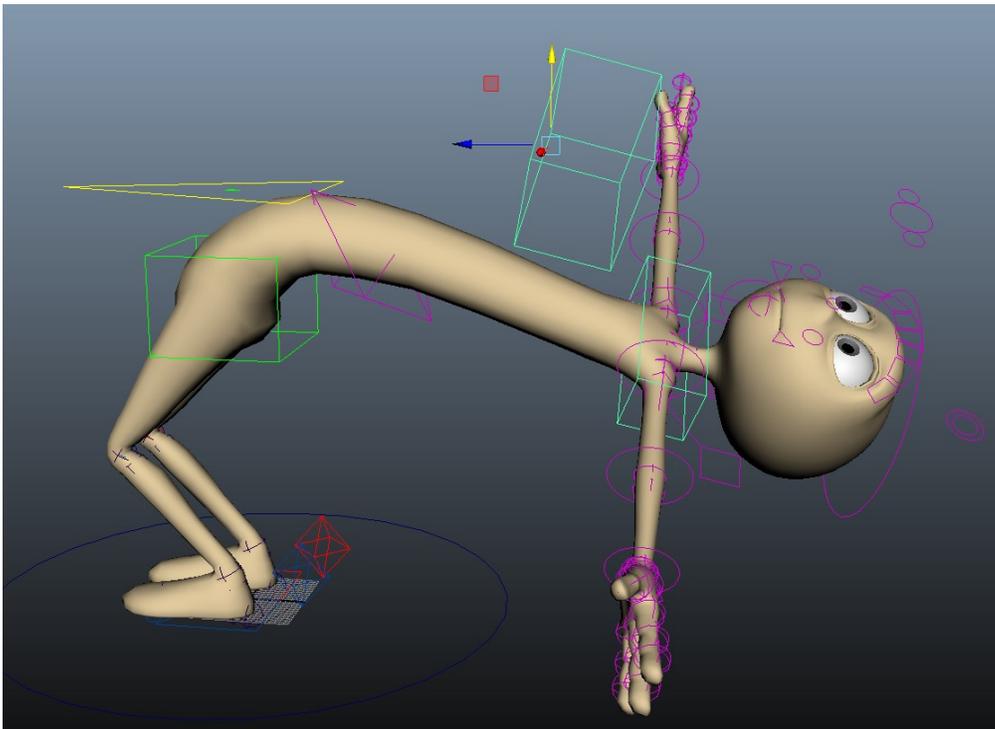


Figure 2.7 Animation Mentor Bishop IK spine rig.

The Bishop spine requires the blending between its FK and IK rigs going against the principle of physical Direct Manipulation.

### **2.5.2 Neck & Head**

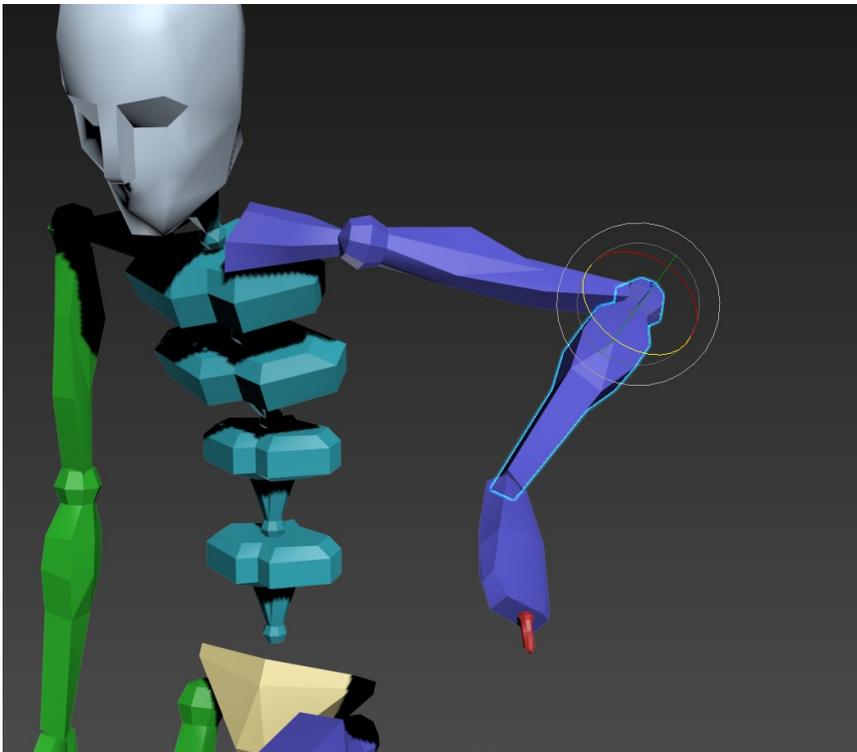
The neck and head should be able to be rotated independently of the rest of the body. This is important as the head leads the actions of the character by turning to face a particular point in space and staying in the orientation, which is vital for communicating intent and building the illusion of consciousness in a character. The Biped rig has an independently rotating head by default and Bishop's neck and head can be switched between dependent and independent rotation. However, the dependent and independent rotation rigs need to be blended between, counter to the principle of physical Direct Manipulation.

### **2.5.3 Legs & Arms**

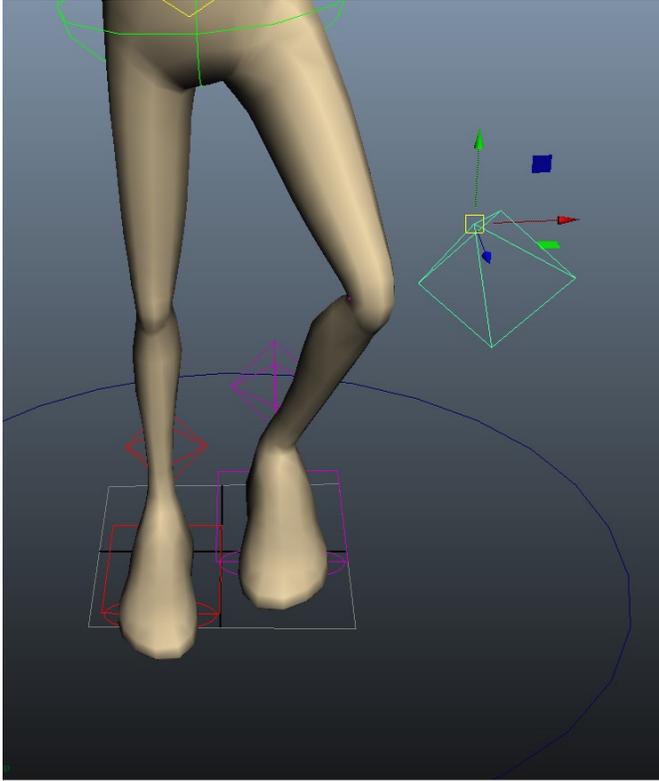
The arms and legs are very similar in requirements, with the exception that both legs are rooted to the same body part – the pelvis – while each arm is attached to its own clavicle, which are similarly dependent/independent in their rotation of the top of the spine as the pelvis is to the bottom. Otherwise, both legs and arms are chains of two bones (three joints) that require a way to be controlled hierarchical by rotation (FK) and by translating the end of the chain (IK). The latter is the more problematic, as it requires the animator to also control where the mid-joint (elbow or knee) should be pointing.

The Biped rig allows for the most intuitive manipulation of this as seen in Figure 2.8, since its bones are simultaneous FK and IK – rotating the upper arm/leg or just translating the elbow/knee directly will give the animator exactly the pose they expect, even if the hand/foot has been locked into a point in space. The Bishop rig, however, only has separate FK and IK sub-rigs and the mid-joint is manipulated by translating a "Point Vector" (PV)

controller and via a numerical "Twist" field as shown in Figure 2.9 neither of which alone is intuitively ideal and, used together, can lead to great confusion as they work against each other. The PV can be animated far off the character model and can lead to all sorts of undesirable animations caused by its unfettered travels through space. The "Twist" parameter is not immediately meaningful and represents the cause of unnecessary cognitive overhead, as the animator has to spend valuable time on getting a feel for what the values actually mean in posing terms.



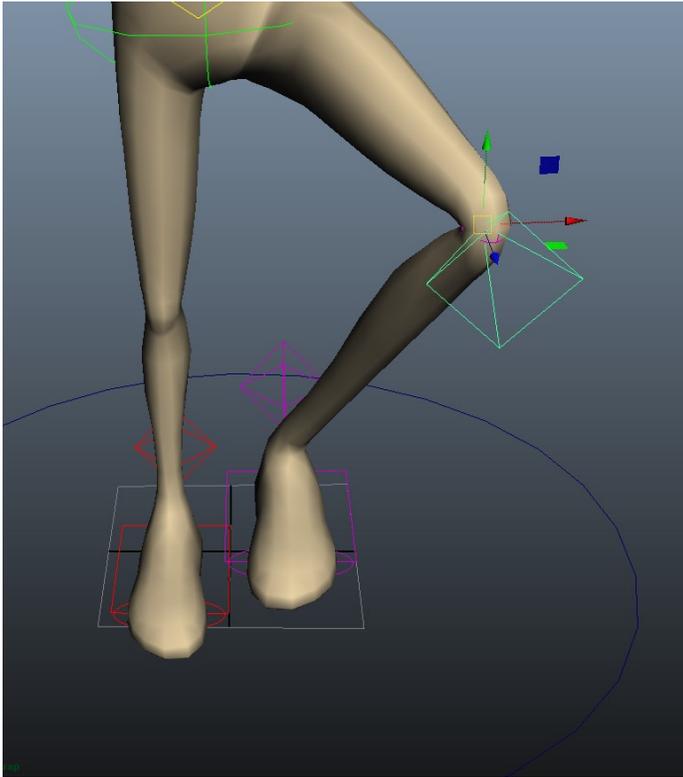
**Figure 2.8 Character Studio Biped arm rig.**



**Figure 2.9 Animation Mentor Bishop PV knee.**

A further complication lies in the need to lockdown the mid-joint, like when characters have to go down on their knees or rest their elbows on a table. Here, the Biped rig falls short, as its elbows and knees cannot be as readily locked onto a plane as its hands and feet. The Bishop rig, however, has mid-joints that can be switched to meet the PV controllers by squashing and stretching as seen in Figure 2.10. While this broadly meets the goal, it does introduce the problem that the animator has to keep a careful eye on the proportions

of the limbs to ensure that they do not squash or stretch beyond the bounds of believability.



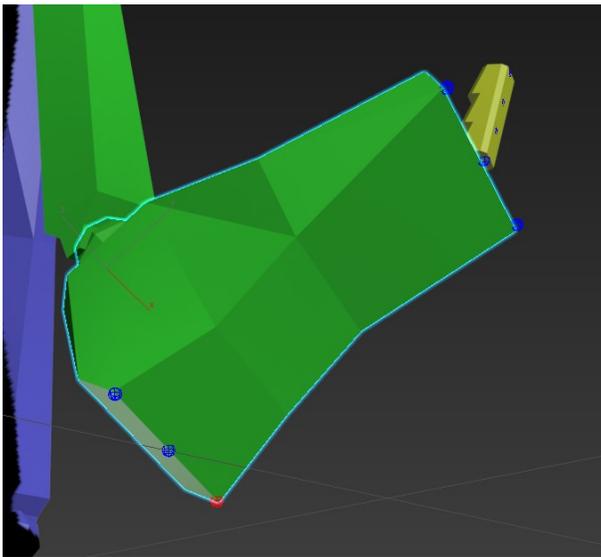
**Figure 2.10 Bishop PV meets knee.**

The Character Studio Biped rig fails at offering full Direct Manipulation control over the mid-point of the limb, and the traditional Bishop rig using blending between IK and FK counter to the physical Direct Manipulation principle. Furthermore, the *rotation* of the IK limb is controlled via a *translation* of the PV controller which goes against the Affordances principle that the controller should give a hint as to control it offers.

#### **2.5.4 IK Feet & Hands**

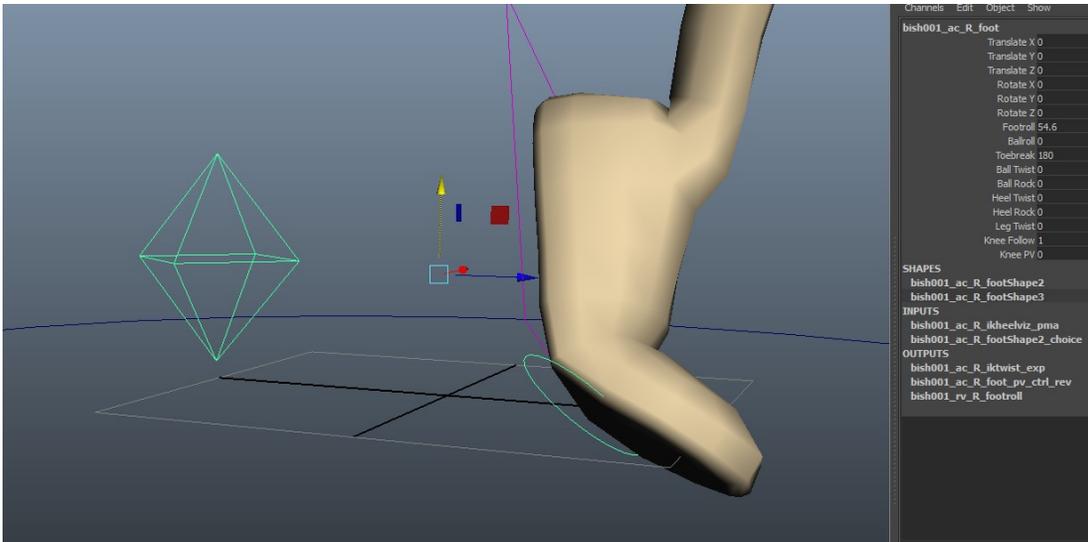
The hands and feet can be the most intricate parts of a bipedal rig as they propel the body forward in a walk and can hold objects with numerous fingers – each finger being as complicated as the spine and potentially even more time-consuming to animate. The Biped rig shines here as seen in Figure 2.11 – the IK feet and hands have animatable shifting pivots extending into the toes/fingers that can easily pivot the whole foot/hand along

surfaces in a pleasingly intuitive manner. Although, the foot/hand does show the rotation/translation widget to be located at the ankle/wrist rather than the location of the currently active pivot which can cause some confusion as to where the actual pivot is, especially since there are many potential pivots and they aren't always readily visible to the animator. This widget misplacement goes against the principle of Mapping as the widget's location is giving no indication as to what pivot its actually acting from.



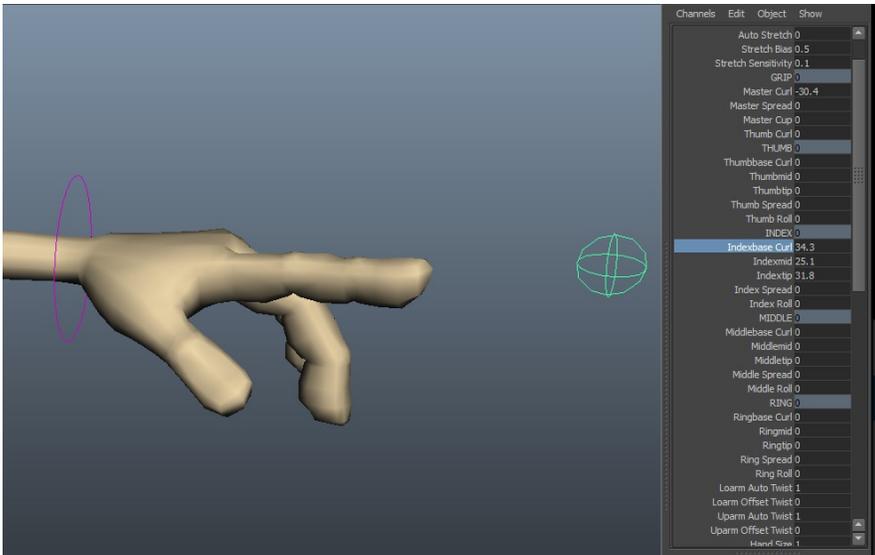
**Figure 2.11 Biped foot – red dot on the heel is pivot point.**

Conversely, the feet and hands are the Bishop's rig greatest failing. The feet as shown in Figure 2.12 are optimized for a heel-ball-toe tip rocking motion on a sandal-like controller that they rock off of and then have to be brought back to if the whole foot is to be moved in any clean manner, since any ball or toe tip pivot motion of the foot take the foot off of its main controller. It leads to all the problems of cascading motion that are the hallmark of hierarchical pivot controllers. First, when motion on a higher-in-the-hierarchy pivot has to be made the foot has to be returned to the neutral pose for that controller which leads to labour-intensive counter-animation. Secondly, any tweaking of pivot animation will lead to undesirable changes to the animations of pivots down the hierarchy.



**Figure 2.12 Bishop foot lifting away from its main pivot.**

The Bishop hands suffer their own hierarchical controller structure problems in the fingers as shown in Figure 2.13. Numerical field parameter controllers are used to control all the fingers at once which may seem like a time-saving measure but can lead to the animator than having to work against those same controls just to achieve a specific motion on each finger. This is particularly taxing if the animator is trying to achieve the illusion of contact with a prop without the fingers going straight through a supposedly solid object. What seems like a time-saver can ultimately become a time-sink.



**Figure 2.13 Bishop digits and their controlling numerical fields.**

The IK hands and feet of the Bishop rig use abstract numerical controls instead of physical controllers, thus going against the Direct Manipulation principle, The feet also go against the Mapping principle by having controllers that don't align with the part of the foot they control – the controller for the whole foot, for instance, is at the heel.

## Chapter 3: Multiple Dynamic Pivots Rig

The existing 3D character rigs are either dependent on particular 3D authoring tools and lacking in features or are based on a traditional industry aggregate design that goes against the relevant HCI principles in the following way:

- Direct Manipulation – requiring blending between rigs like FK and IK does not adhere to the physical manipulation principle since the character is actually being controlled by the abstract rate of blending value between two rigs
- Affordances – the pivots of the controller widgets do not consistently align with the action pivot of the motion being caused
- Mapping – the position of controllers does not consistently give an accurate clue as to what part of the character they will manipulate

The Multiple Dynamic Pivots (MDP) Rig design discussed in this thesis will remedy all the above issue by leveraging the natural fit of the animation workflow to the HCI principles of Direct Manipulation, Affordances and Mapping.

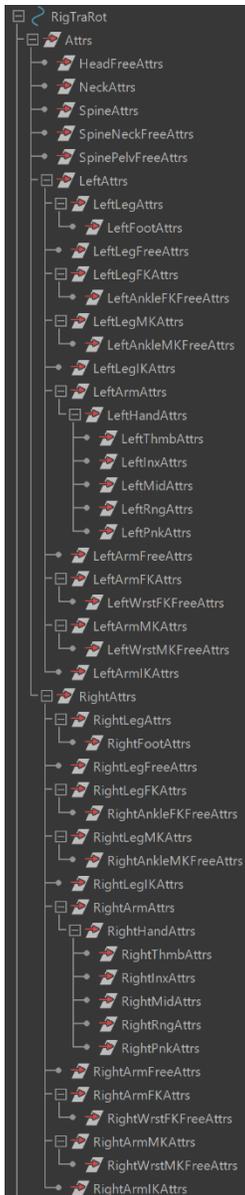
The core mechanic of the MDP Rig is its ability to pivot the spine and limbs from multiple points within each of those bodily structures. This is achieved with various sub-rigs for each body part – each sub-rig having their own unique pivot – that align to each other at the press of a button. This "auto-alignment" switch from one "mode" (sub-rig) to another consists of the position, rotation and scale of the active sub-rig's controllers to all be keyed on the current frame, and the new mode's controllers to be positioned, rotated and scaled in such a way that the joints they control are aligned to the joints of the previous mode on the same frame. Similarly, on such a switching keyframe, the previous mode's controllers can be aligned and keyed to the active mode's controllers without switching back to it –

this ensures the smooth playback of the animation over the switch frame as shown in Figure 3.1.



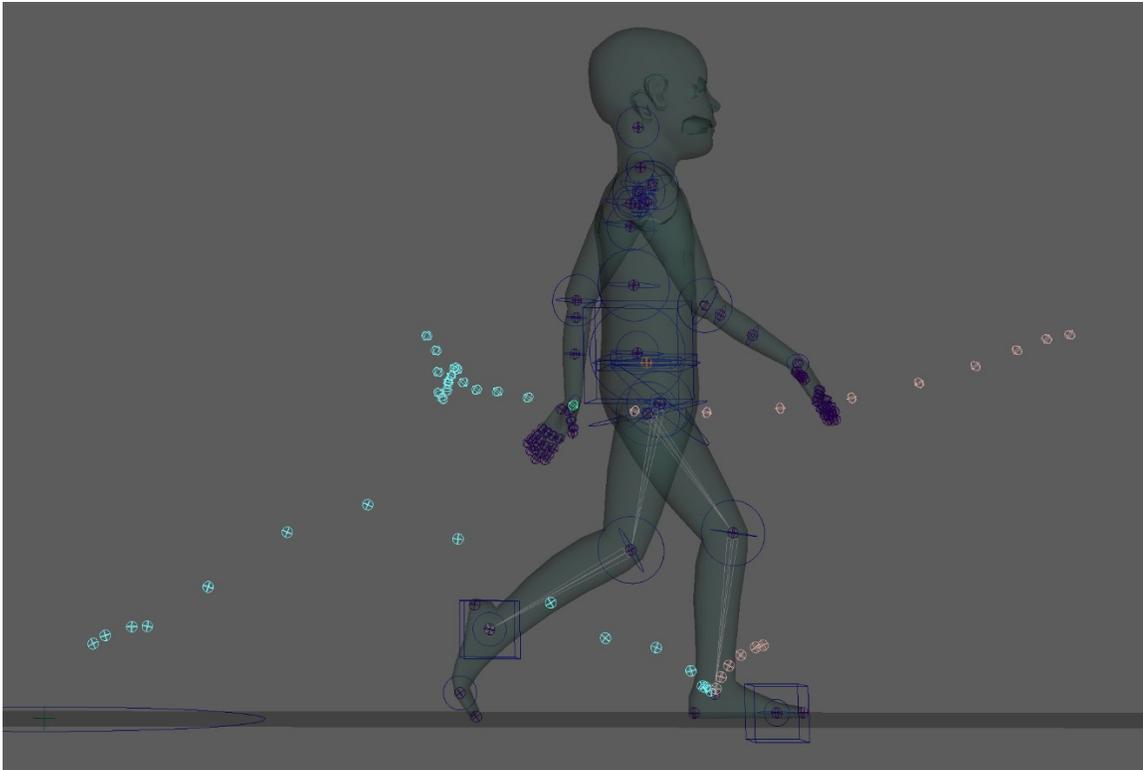
**Figure 3.1 The MDP IK Foot switching pivot modes from Ball (green) to Toe tip (pink) to Ankle (blue) to Heel (tan) and back to Ball in a walk animation.**

The converse consideration of the MDP rig's automation power is that it must not come at the expense of the animators' power – animators must be able to see and edit the results of the automatic processes and undo those results if they wish (Nielsen, 1994a) (Nielsen, 1994b) (Blumethal, 2016). Thus, the mode switching attributes that are accessible within the rig's structure as shown in Figure 3.2 can be edited, keyed and viewed in the graph editor. The "Skin" joints that are attached to the geometry and controlled by the MDP rig are unhide-able, should the animator choose, and can be given motion trails as a visual aid to creating those much-sought motion arcs and slow ins and outs as shown in Figure 3.3.



**Figure 3.2** The mode switching attributes of the MDP rig organized in a logical hierarchy under the rig's master controller.

The animator is also free to delete the mode switching instantly by initiating an "undo" command or by deleting the mode switch via the rig UI. Never should animators be made to feel that the MDP rig is taking control away from them.

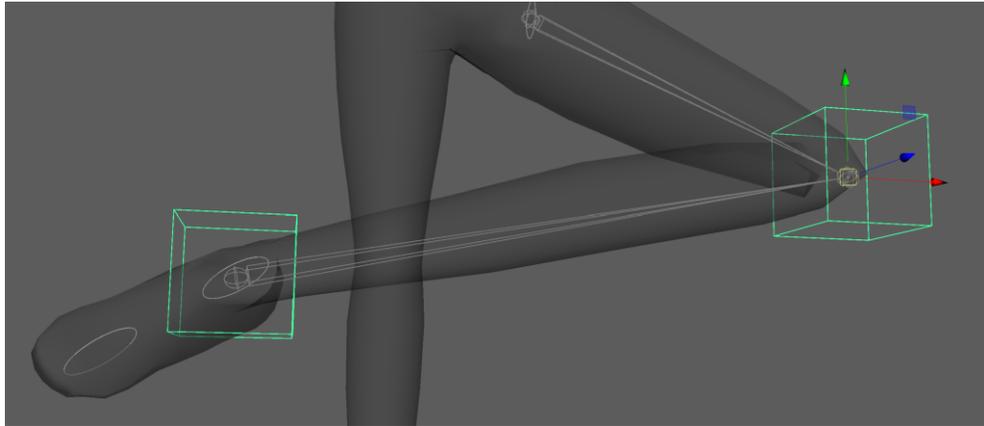


**Figure 3.3 The motion trails of the left wrist and right ankle during a walk cycle – the blue trail is in the past and the pink in the future of the current frame.**

Unfortunately, having separate controllers for each transform type would cause controller over-crowding around the pivots and make them difficult to select, thus going against Fitts's Law (Fitts, 1954) (MacKenzie, 1992). As a compromise, most of the rotation controllers of the MDP rig can also manipulate scaling. The specific diamond scaling controller is only present in the IK limbs, where there are no rotation controllers to also handle length scaling along the limb segment.

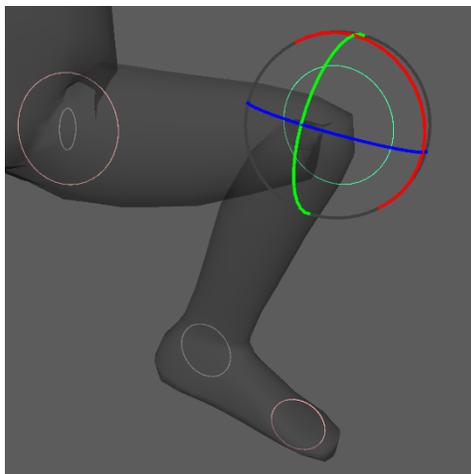
The appearance of the controllers themselves was a secondary priority in designing the MDP rig. The controllers are kept as simple as possible while still giving some visual cue as to their use in accordance with Norman's UI principle of Affordances and Mapping (Norman, 2013). Affordances-wise, the shapes of the controllers are indicative of their primary manipulation capability:

- Cubes are used for "-Tra" suffix positional controllers as shown in Figure 3.4 – their 3D angularity similar to the translate widget signifying that they can be translated along all axes.



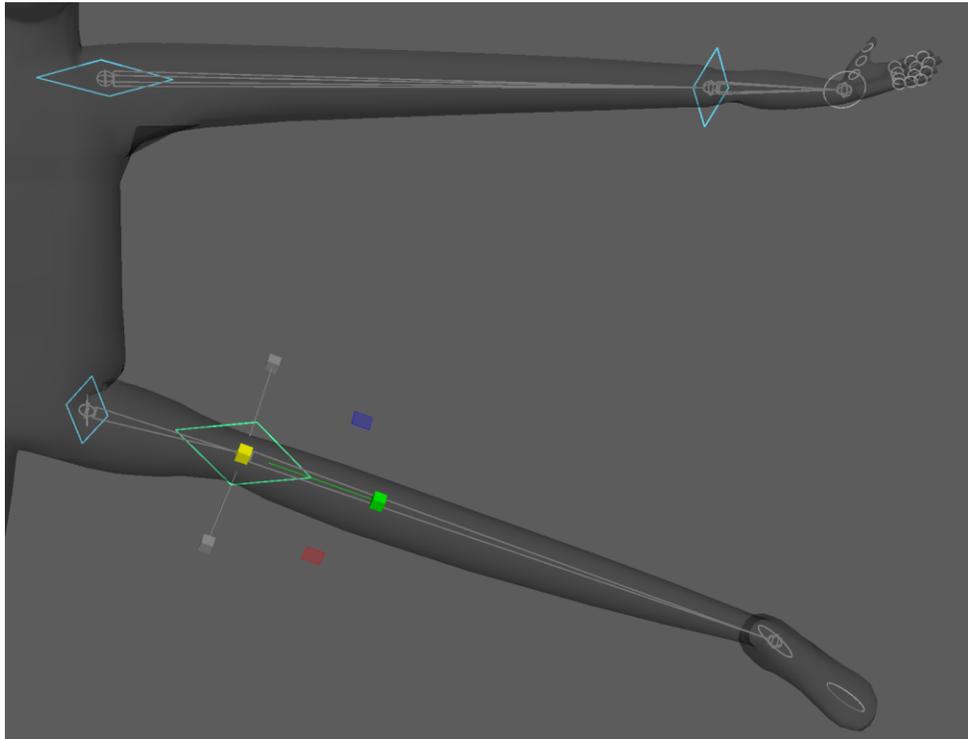
**Figure 3.4 Double IK leg translation controllers with translate widget active on knee.**

- Circles are used for "-Rot" and "-Lift" suffix rotational controllers as shown in Figure 3.5 – their 2D roundness similar to the rotate widget signifying that they can be rotated primarily, but not necessarily exclusively, about the axis they encircle.



**Figure 3.5 Knee FK rotation controller with rotate widget active.**

- Diamonds are used for "-Sca" suffix scaling controllers as shown in Figure 3.6 – their 2D angularity indicating, rather by process of elimination in the context of the other controller shapes, that they're neither 3D movement or rotation controllers so they must be the only remaining transform type: scaling.



**Figure 3.6 IK limb scaling controllers with scale widget active on knee.**

The position and alignment in space of the controllers is also significant in accordance with the Mapping principle. The controllers are all centred on the pivot they control, and their alignment is just as significant:

- Cube translation controllers are by default aligned to the World space to indicate that their rotation does not affect the pivot directly.
- Circle rotation controllers are aligned in Local space to encircle their intended primary axis of rotation so that when the rotate widget is active on them, their shape aligns with their primary rotation axis on the widget.

- Diamond scale controllers are aligned in Local space in the direction of the segment length scaling they are capable of.

Unfortunately, having separate controllers for each transform type would cause controller over-crowding around the pivots and make them difficult to select, thus going against Fitts's Law (Fitts, 1954) (MacKenzie, 1992). As a compromise, most of the rotation controllers of the MDP rig can also manipulate scaling. The specific diamond scaling controllers are only present in the IK limbs where there are no rotation controllers to also handle length scaling along the limb segment.

### **3.1 Rig Controller**

The whole MDP rig can be positioned and rotated into its starting place in the Cartesian space of the 3D scene by the RigTraRot controller. Once placed, the RigTraRot controller is not designed to be moved during the animation of the scene. Functionally, the RigTraRot is where the translation controllers of the rig will go back to in their start positions and rotations if those values are all set to zero. Translation controllers only dictate the positions of their respective rig parts in relation to the RigTraRot but they have a keyable rotation to allow for the Local space swerving of their trajectory through 3D space, should the animator need it to create arcing paths of motion.

### **3.2 Spine**

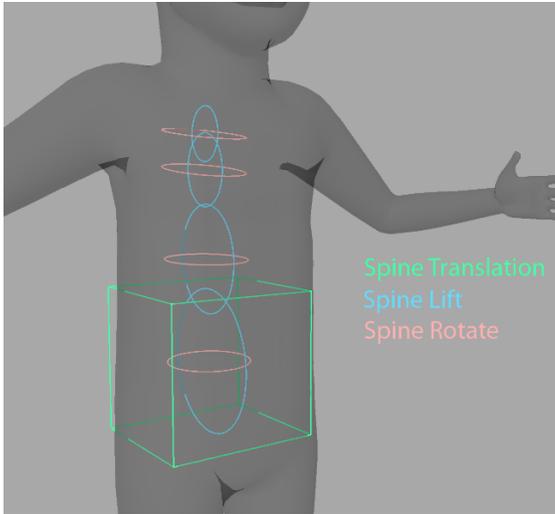
The MDP spine pivots from 4 points along the torso from the pelvis to the base of the neck as shown in Figure 3.7 to 3.10. The spine sub-rigs are all variations on the spine vertebrae hierarchies consisting of the following controllers:

- Translate - moves the whole spine.

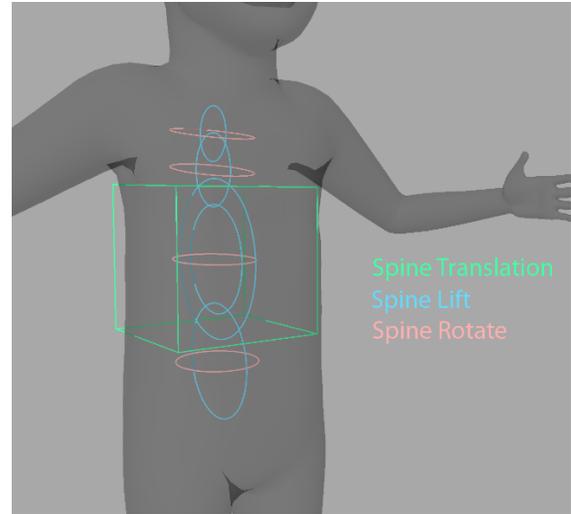
- Lift - orbits a neighbouring joint about the joint the controller is centred on and scales the distance between joints and the thickness of the joint it's on.
- Rotate - rotates the joint the controller is centred on without affecting neighbouring joints.

The Spine Translation controller of each sub-rig can move and rotate the whole body as long as the limbs are in Forward Kinematics mode. Likewise, the Spine Lift controller of the sub-rigs that pivot at pelvis and neck – Spine1 and Spine4, respectively – will rotate the whole body as long as the pelvis and neck are in Follow mode. The Spine2 and Spine3 sub-rigs, however, have a chain of Lift controllers going up and down the spine from the pivot point so both the Lift controllers on the pivot have to be rotated to rotate the whole body.

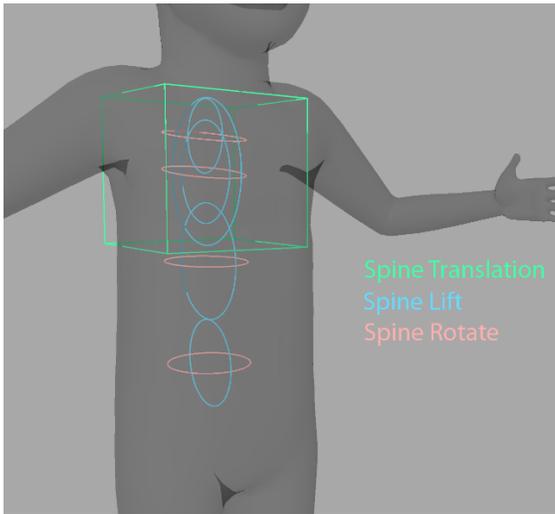
The MDP spine is designed to radiate torso contortions from any of the 4 points along it as shown in Figure 3.11 to 3.14. This is useful for when the torso is against a barrier – like the ground or a bar – and its joints need to rotate out from the point of contact. Also, when the centre of mass of the body is shifted from the pelvis to up the torso – like if the character is walking on their hands. The same design consideration applies to the scaling so that the spine segment can be squashed in and stretched out from any pivot-point along the spine as shown in Figure 3.15.



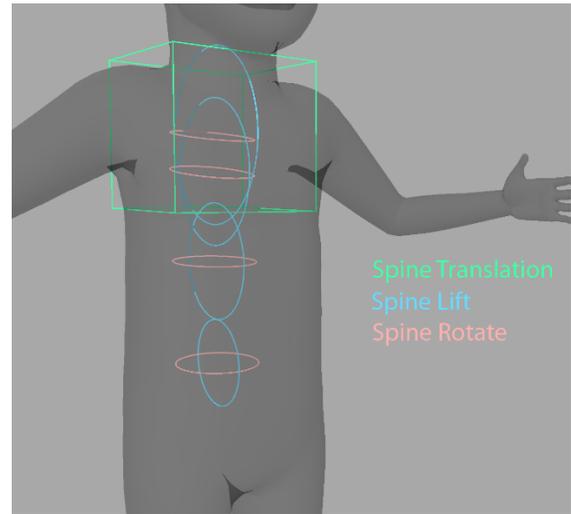
**Figure 3.7 Controllers of Spine1.**



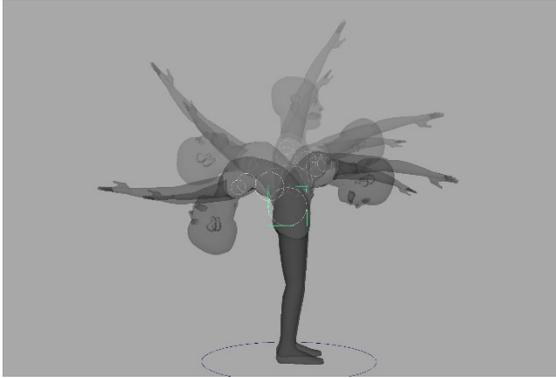
**Figure 3.8 Controllers of Spine2.**



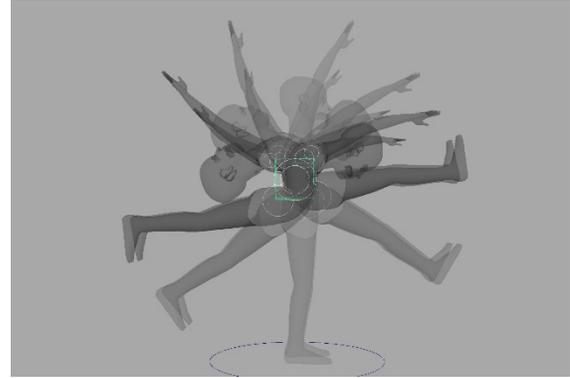
**Figure 3.9 Controllers of Spine3.**



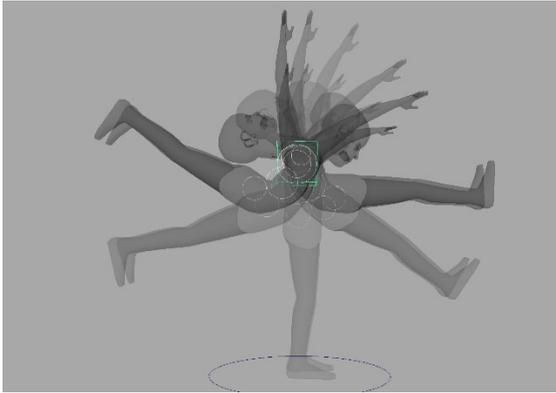
**Figure 3.10 Controllers of Spine4.**



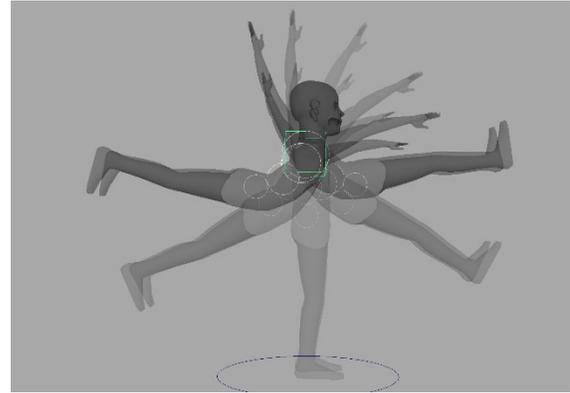
**Figure 3.11 Motion from Spine1 pivot.**



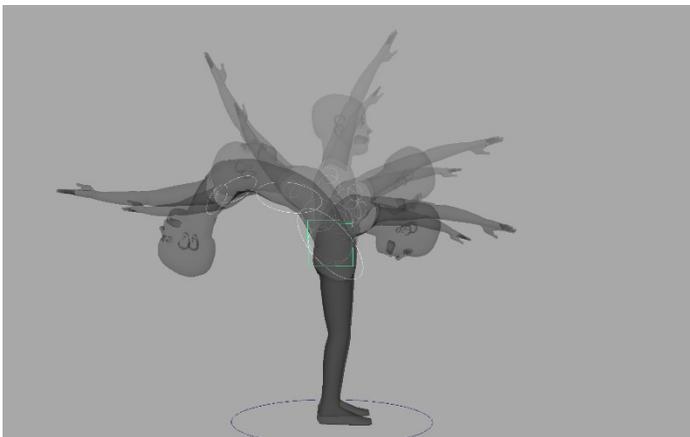
**Figure 3.12 Motion from Spine2 pivot – lower and upper hierarchies rooted at pivot move independently of each other.**



**Figure 3.13 Motion from Spine3 pivot – lower and upper hierarchies rooted at pivot move independently of each other.**



**Figure 3.14 Motion from Spine4 pivot.**



**Figure 3.15 Motion from Spine1 pivot with stretch (left) and squash (right).**

### 3.3 Limbs

The MDP limbs have 4 modes each:

1. Forward Kinematic (FK)
2. Mid-joint Kinematic (MK)
3. Inverse Kinematic (IK)
4. Double Inverse Kinematic (DK)

#### 3.3.1 FK

The Forward Kinematics (FK) mode shown in Figure 3.16 is the simplest limb sub-rig consisting of a chain of rotation controllers – top joint (shoulder/hip) parent of middle joint (elbow/knee) parent of end joint (wrist/ankle) as shown in Figure 3.17 The top and middle rotation controllers also scale the length of each limb segment as in Figure 3.18.

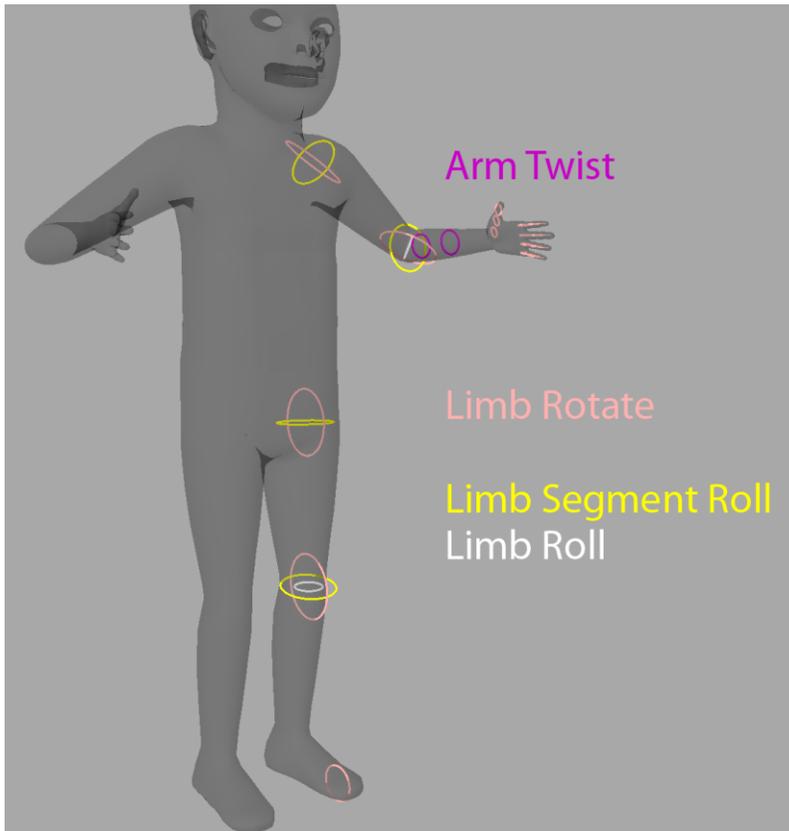
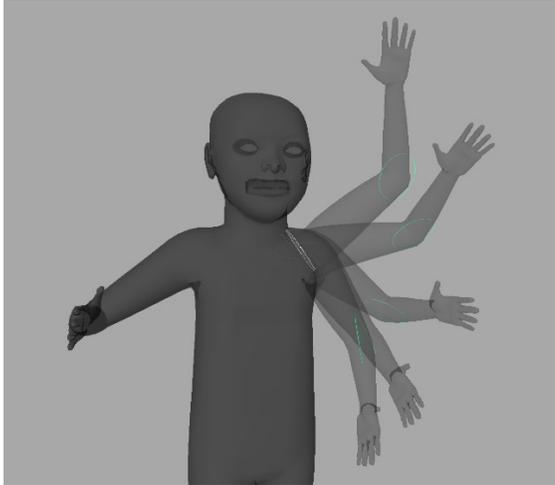
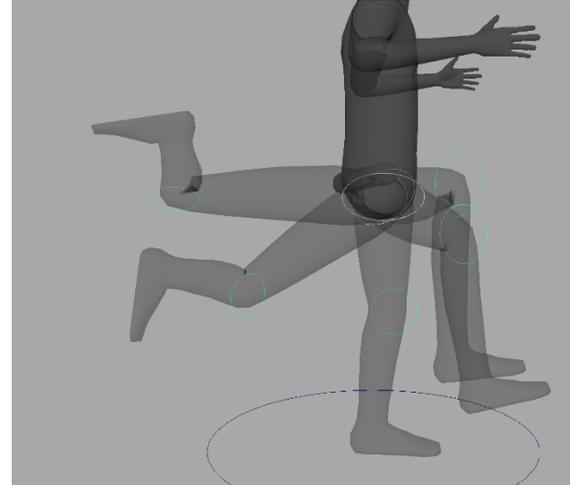


Figure 3.16 Controllers of the FK limbs.



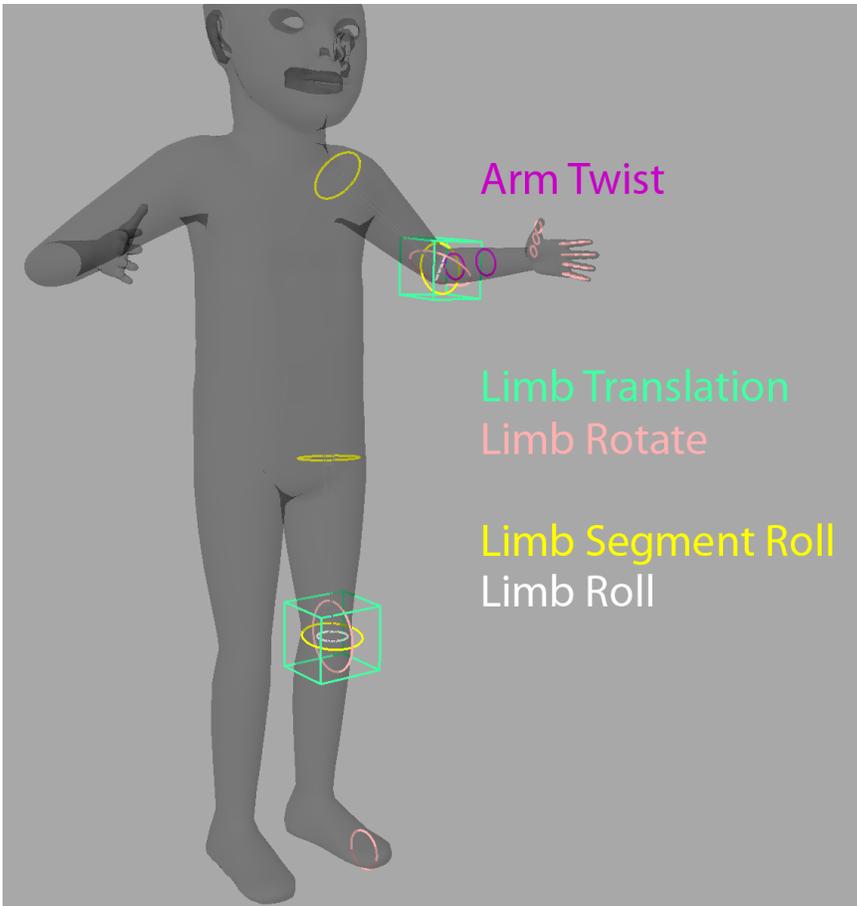
**Figure 3.17 Motion of the FK arm.**



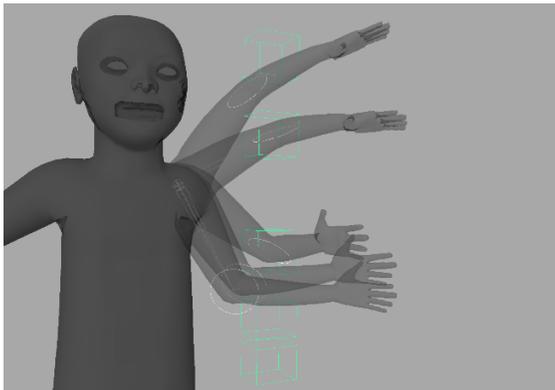
**Figure 3.18 Motion of the FK leg with upper-bone stretch and lower-bone squash (left) and upper-bone squash and lower-bone stretch (right).**

### **3.3.2 MK**

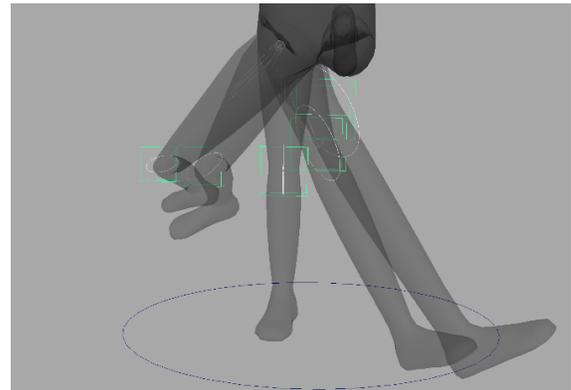
The Mid-joint Kinematics (MK) mode shown in Figure 3.19 allows the planting of the mid-joint via a middle translate controller and a middle rotate controller for the lower limb segment rotation as shown in Figure 3.20. The top segment length is optionally squashed and stretched by the mid-joint actually sticking to the translate controller exactly instead of just pointing toward its position when the squash & stretch is disabled. The lower segment is scaled in length by the middle rotation controller like in the FK mode as in Figure 3.21.



**Figure 3.19** Controllers of the MK limbs.



**Figure 3.20** Motion of the MK arm.



**Figure 3.21** Motion of the MK leg with upper-bone stretch and lower-bone squash (left) and upper-bone squash and lower-bone stretch (right).

### 3.3.3 IK

The Inverse Kinematics (IK) mode shown in Figure 3.22 allows the planting of the end-joint via an end translate controller and the swiveling of the whole limb about the top-to-end pole shown in Figure 3.23 via a limb rotation controller located at the top point as in Figure 3.24. This IK rotation controller replaces the convention Point Vector translation controller to make the act rotating the limb be more intuitively matched by the act of rotating a controller in accordance with Norman's principle of Affordances (Norman, 2013). The optional stretching of the whole limb is accomplished by both segments lengthening, so that the end-joint meets the exact position of the translation controller and squashing is accomplished by both segments shortening to form a straight joint line between the top and end points of the limb as shown in Figure 3.25. Stretch can be enabled without squash being enabled, but not vice versa. The IK scale controllers at the top- and mid-joint scale the length of the upper and lower segment, respectively, regardless of whether the limb's squash and stretch are enabled or not. They also allow the independent scaling of each segment in addition of the squash and stretch.

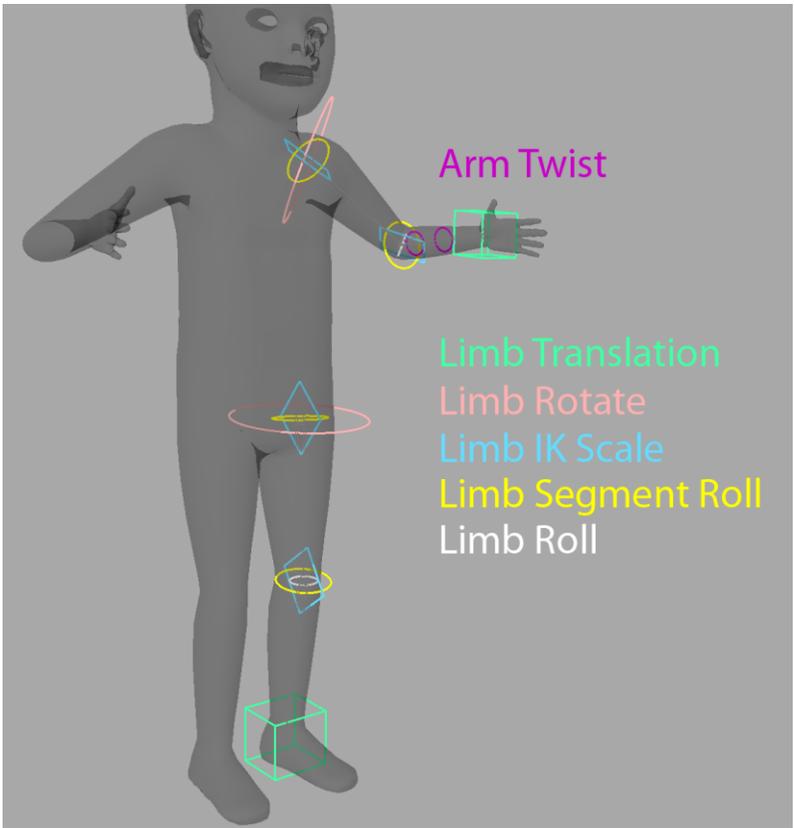


Figure 3.22 Controllers of the IK limbs.

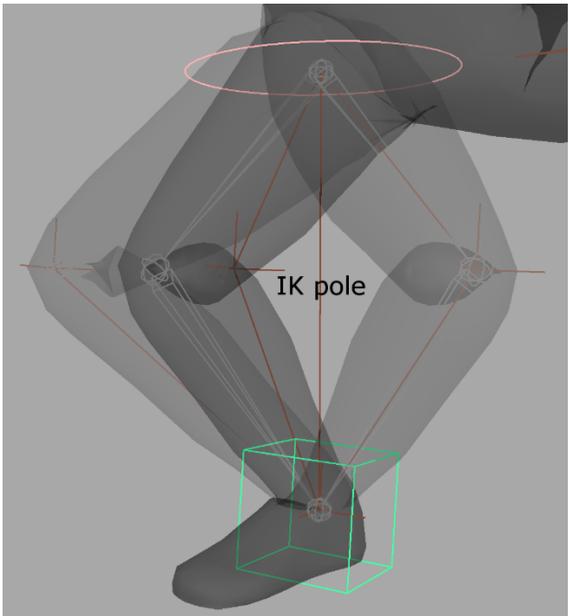
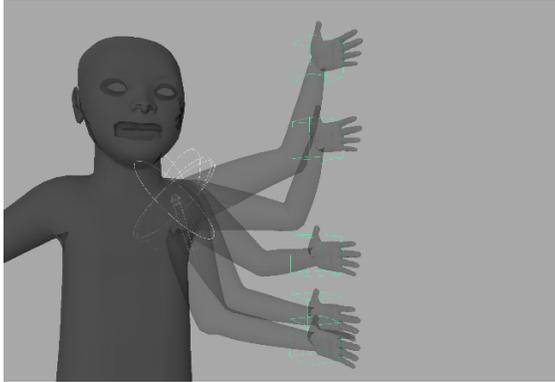
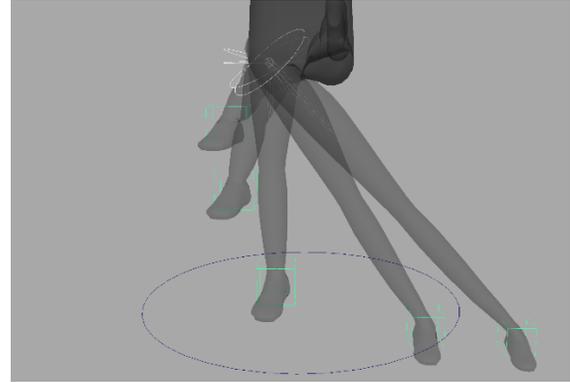


Figure 3.23 The IK pole is the straight line between the top and end joints of the IK chain – the pole that the IK chain rotates about.



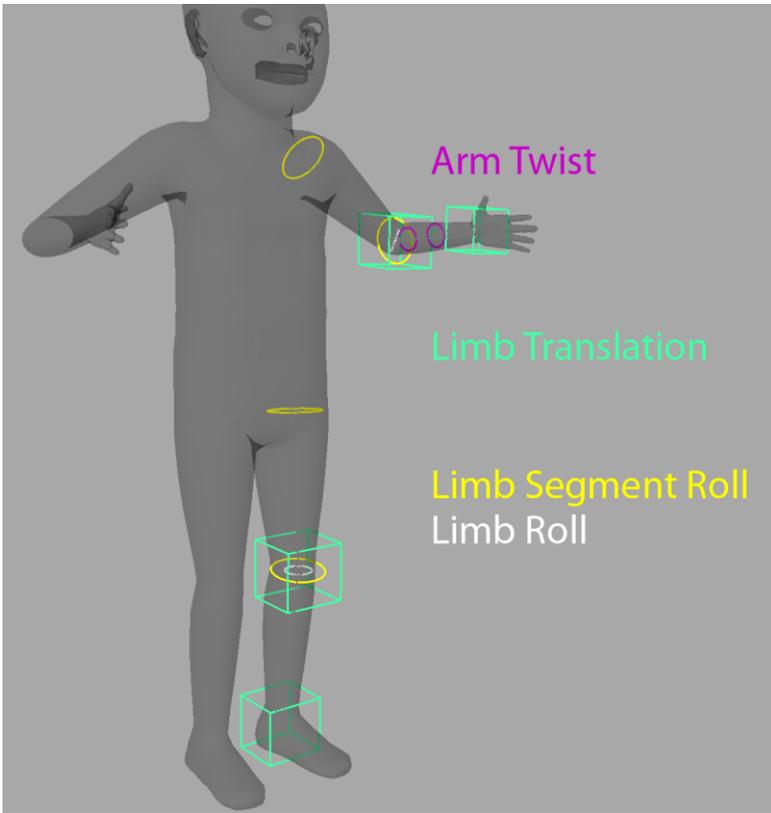
**Figure 3.24 Motion of the IK arm.**



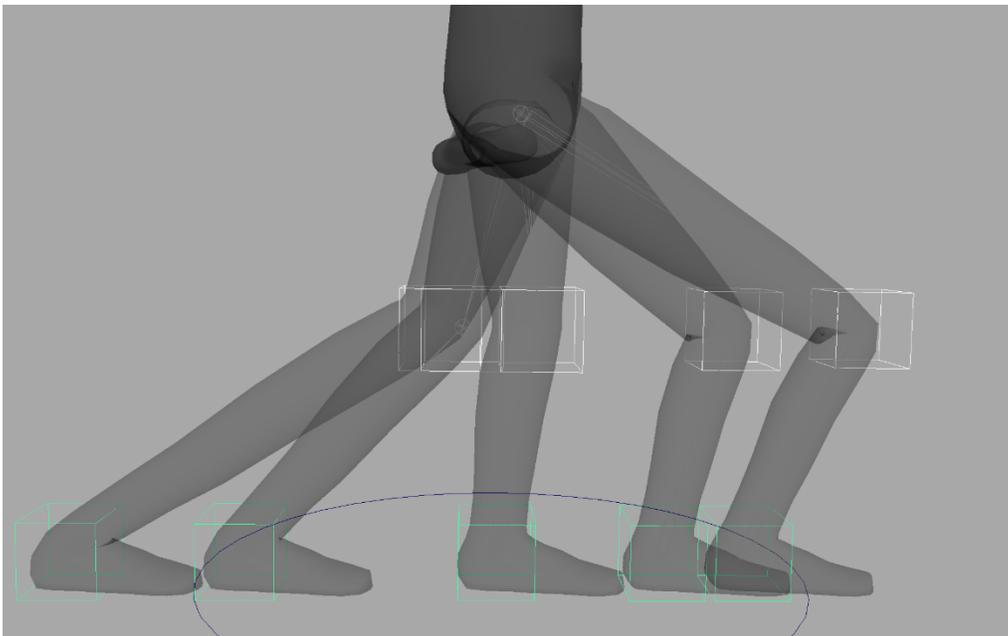
**Figure 3.25 Motion of the IK leg with whole limb squash (left) and stretch (right).**

### **3.3.4 DK**

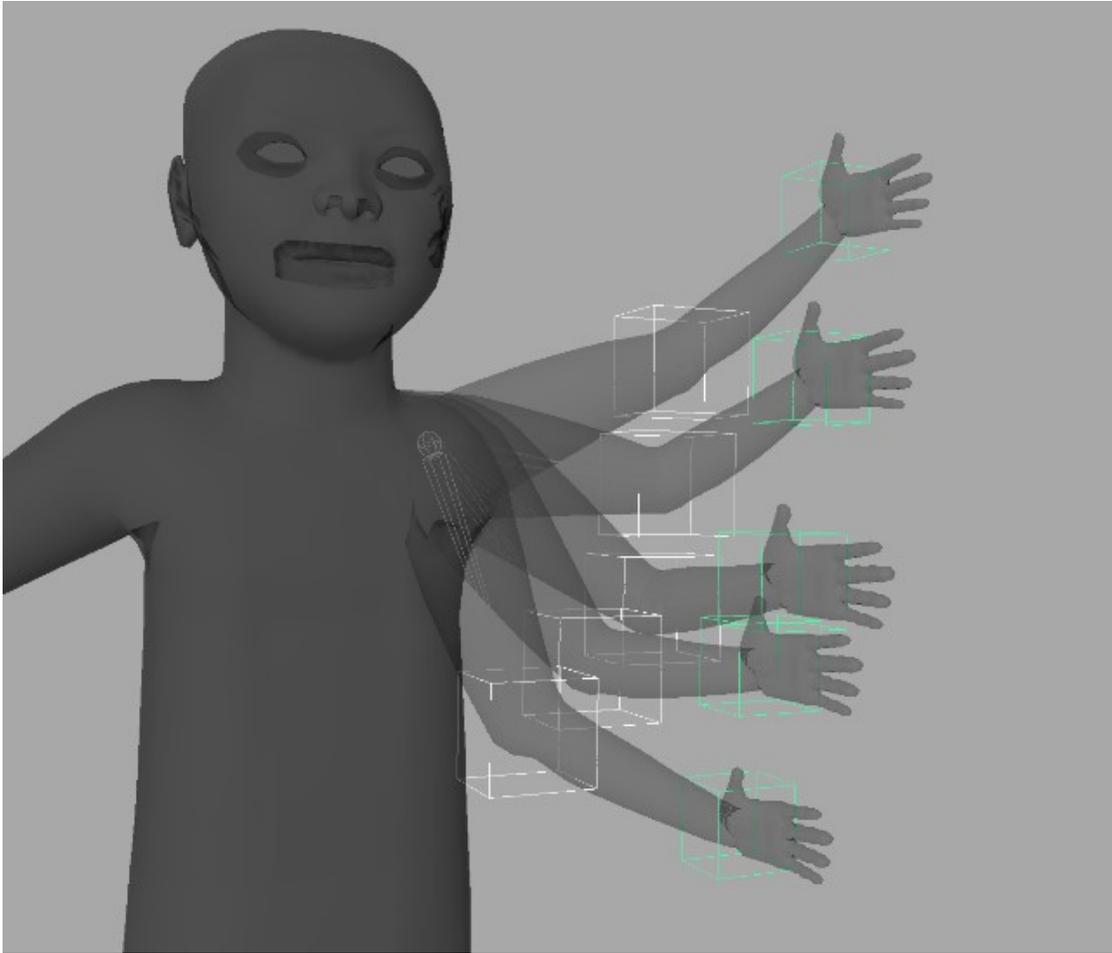
The Double Inverse Kinematics (DK) mode shown in Figure 3.26 allows the planting of the mid-joint and the end-joint independently of each other via a middle and end point translation controller, respectively. The squashing and stretching of the limb segment lengths is not optional in the DK mode as its purpose is to fix the joints in space as seen in Figure 3.27. However the length of the segments can be reset to default with the press of a button causing the DK end translate controllers – each has its own reset button – to move back to their initial length position as shown in Figure 3.28.



**Figure 3.26 Controllers of the DK limbs.**



**Figure 3.27 Motion of the DK leg with upper-bone squashed and lower-bone stretched (left) and upper-bone stretched and lower-bone squashed (right).**



**Figure 3.28 Motion of the DK arm with length of the bones resets at the extreme poses.**

### **3.3.5 Roll**

The limbs have universal roll controllers that aren't part of any daughter sub-rig for each segment that rotate the top- and mid-joints about the length axis of each segment. These controllers also handle the thickness scaling of each joint on the two local lateral axes. The whole limb roll controller mitigating the "candywrapping" of the geometry mesh about the rig when the limbs do extreme rotations as shown in Figure 3.29 and 3.30. It does not handle geometry scaling.



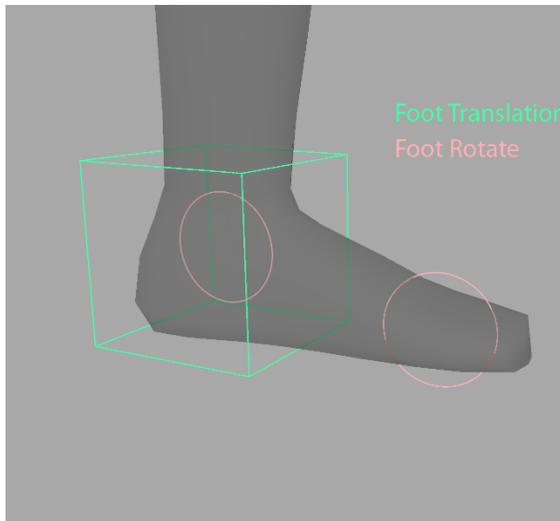
**Figure 3.29 Candywrapping of the geometry mesh as shoulder, hip and ankle.**



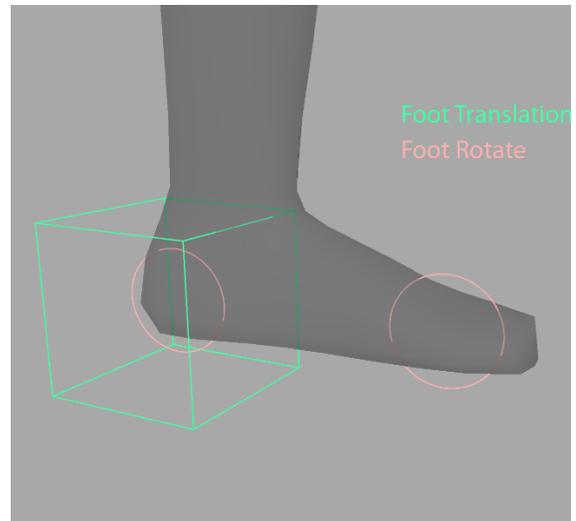
**Figure 3.30 Candywrapping of the geometry mesh as shoulder, hip and ankle mitigated by the rotation of the limb roll controllers.**

### **3.4 IK/DK Feet**

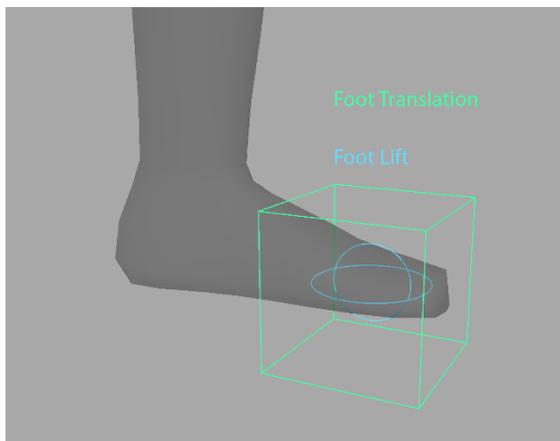
The MDP IK/DK foot moves the IK and DK end-joints attached to it with 4 possible pivot modes: ankle, heel, ball of foot and toe tip as shown in Figure 3.31 to 3.34. Each mode has a whole foot translation controller and rotation controller on its respective pivot point, with a rotation controller on the ball. The ball rotator swivels the toe in the ankle, heel and ball modes, and the rest of the foot except the toe on the toe tip mode as in Figure 3.35 to 3.38.



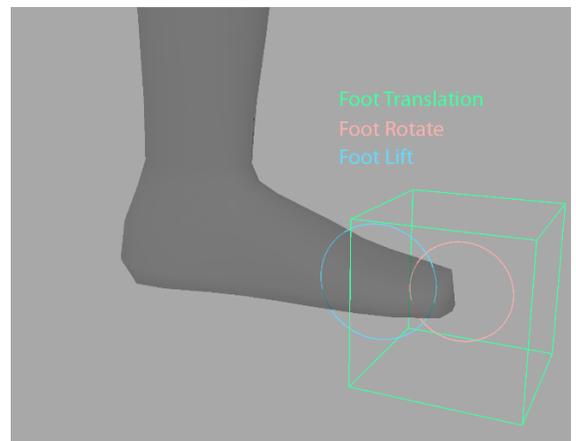
**Figure 3.31** Controllers of the IK/DK Ankle pivot foot.



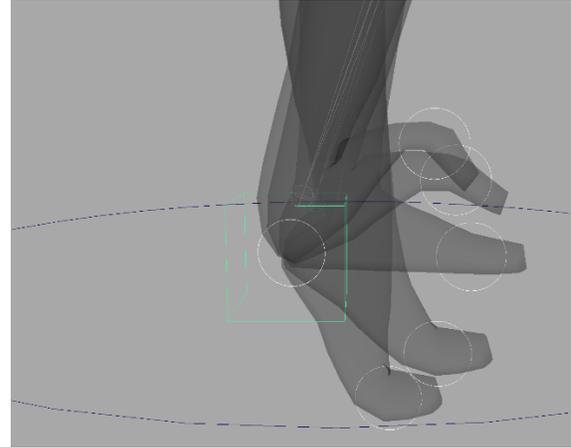
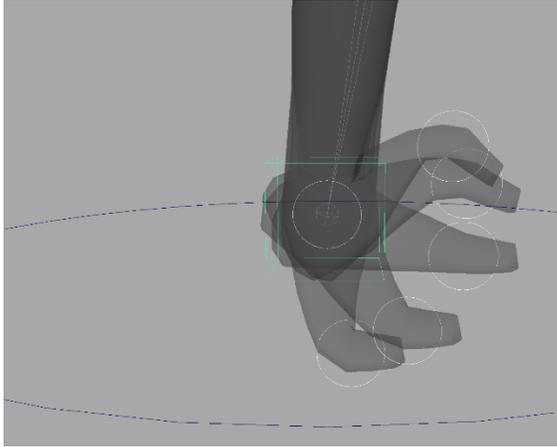
**Figure 3.32** Controllers of the IK/DK Heel pivot foot.



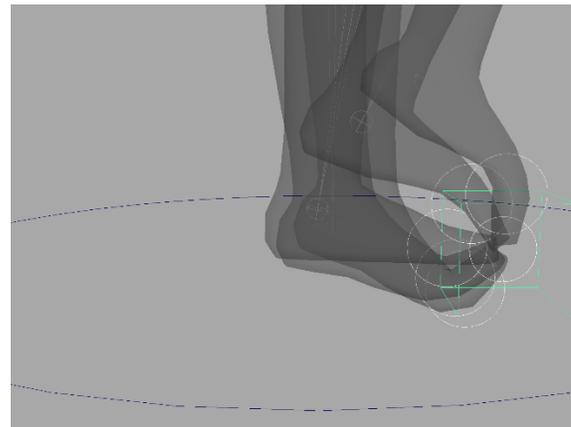
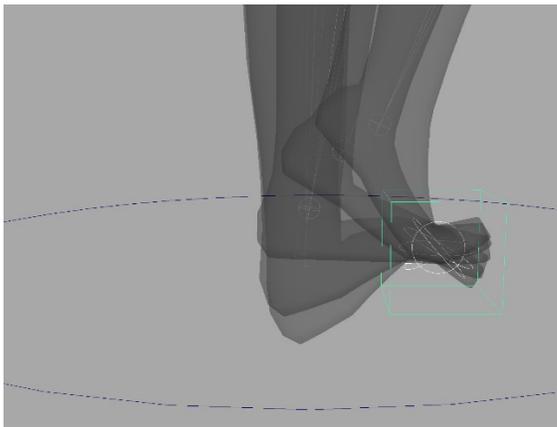
**Figure 3.33** Controllers of the IK/DK Ball pivot foot.



**Figure 3.34** Controllers of the IK/DK Toe tip pivot foot.



**Figure 3.35 Motion of the IK/DK Ankle pivot** **Figure 3.36 Motion of the IK/DK Heel pivot foot.**  
**foot.**

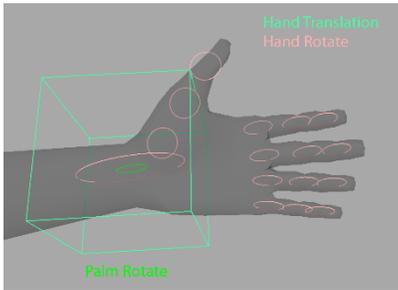


**Figure 3.37 Motion of the IK/DK Ball pivot foot** **Figure 3.38 Motion of the IK/DK Toe-tip pivot**  
**– the heel and toe lift controllers are independent** **foot.**  
**from each other.**

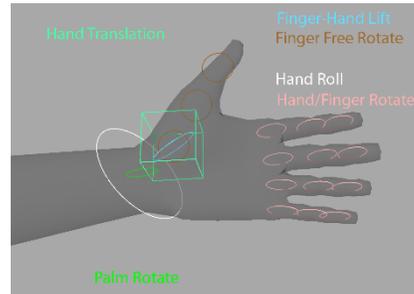
### 3.5 IK/DK Hand

The MDP IK/DK hand moves the IK and DK end-joints attached to it with 21 possible pivot modes – the wrist and every joint plus the tip of all 5 fingers as in Figure 3.39 to 3.43. Each mode has a whole hand translation controller and rotation controller on its respective pivot points as shown in Figure 3.44 to 3.50. The finger joint pivot modes each have a

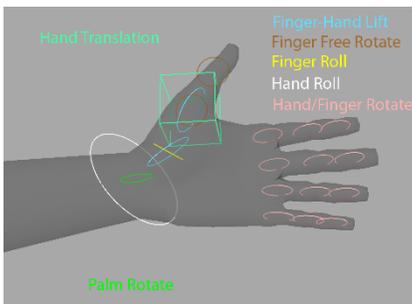
whole hand roll controller, finger lift controllers that chain up the finger from the pivot joint to the wrist with a roll controller that handles each finger segment individually. There are free rotation controllers on each finger joint from the finger pivot point to the tip segment.



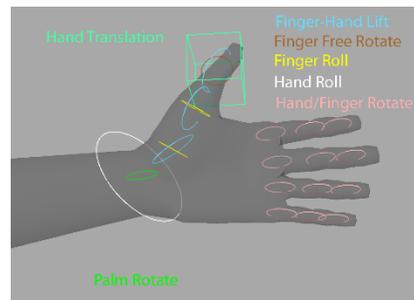
**Figure 3.39 Controllers of the IK/DK Wrist pivot hand.**



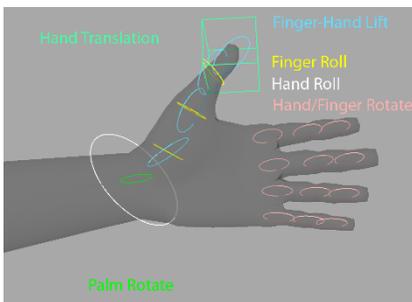
**Figure 3.40 Controllers of the IK/DK Finger1 – Thumb1 in this case – pivot hand.**



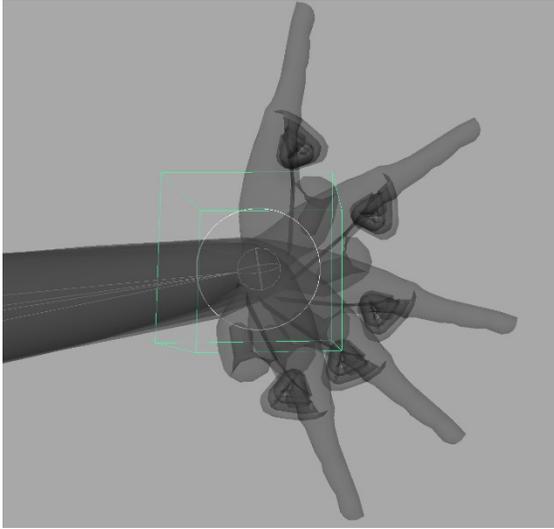
**Figure 3.41 Controllers of the IK/DK Finger2 – Thumb2 in this case – pivot hand.**



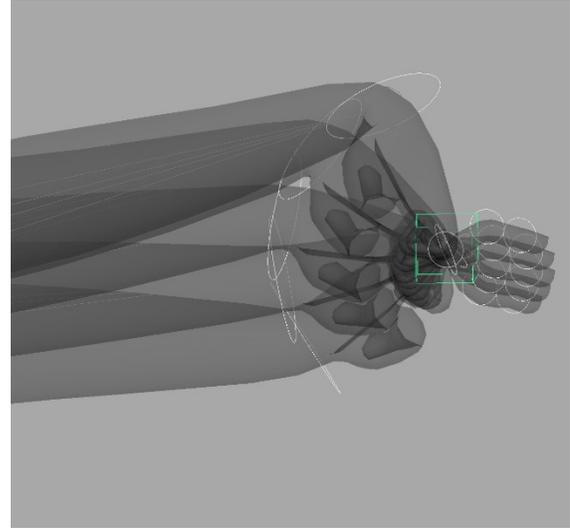
**Figure 3.42 Controllers of the IK/DK Finger3 – Thumb3 in this case – pivot hand.**



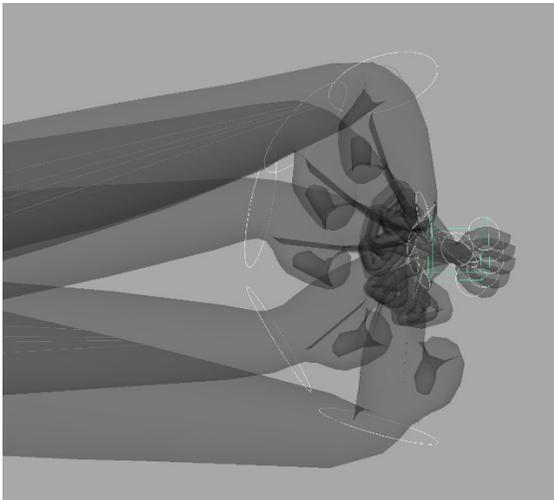
**Figure 3.43 Controllers of the IK/DK Finger4 – Thumb4 in this case – pivot hand.**



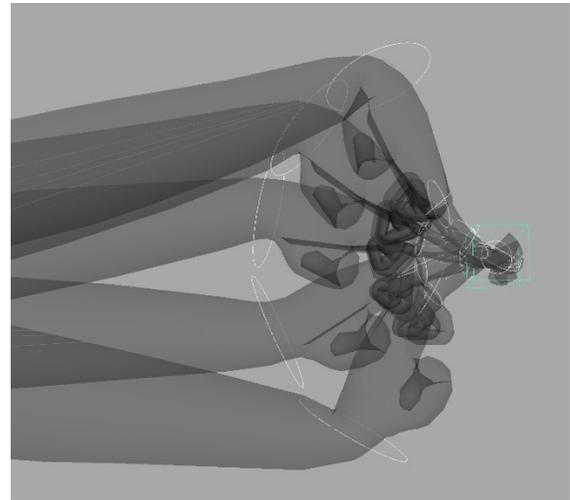
**Figure 3.44 Motion of the IK/DK Wrist pivot hand.**



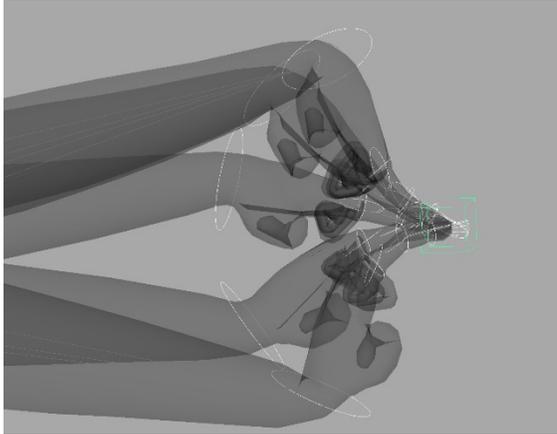
**Figure 3.45 Motion of the IK/DK Finger1 – Index1 in this case – pivot hand.**



**Figure 3.46 Motion of the IK/DK Finger2 – Index2 in this case – pivot hand.**



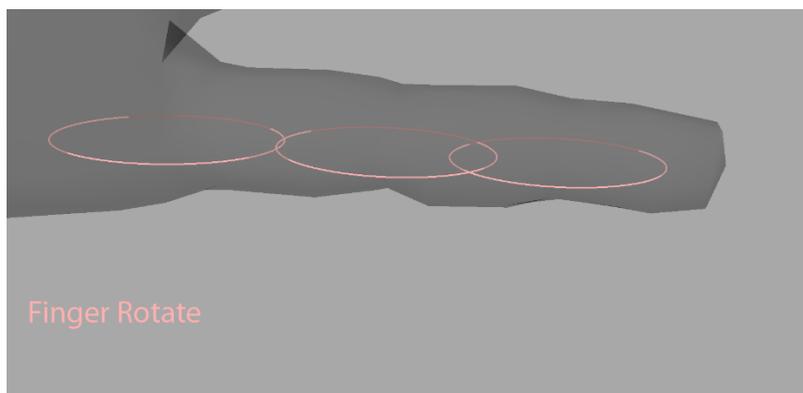
**Figure 3.47 Motion of the IK/DK Finger3 – Index3 in this case – pivot hand.**



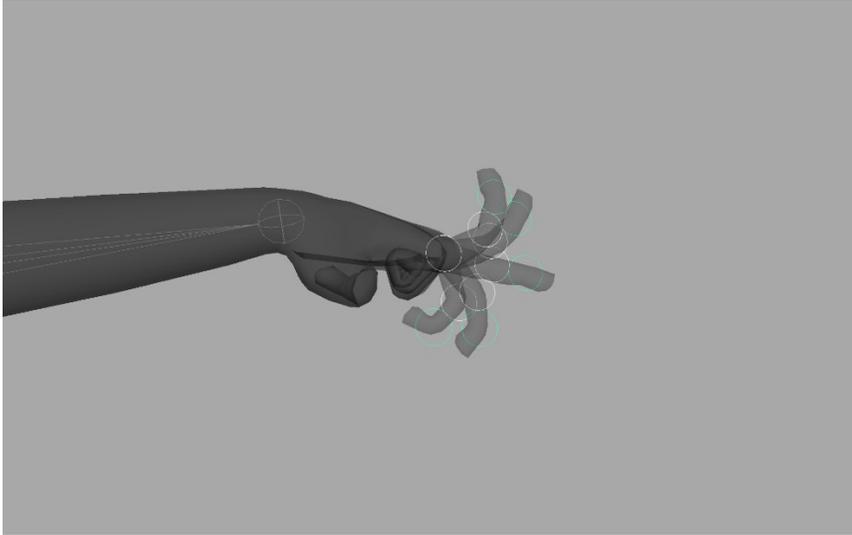
**Figure 3.48 Motion of the IK/DK Finger4 – Index4 in this case – pivot hand.**

### **3.6 Sticky Finger**

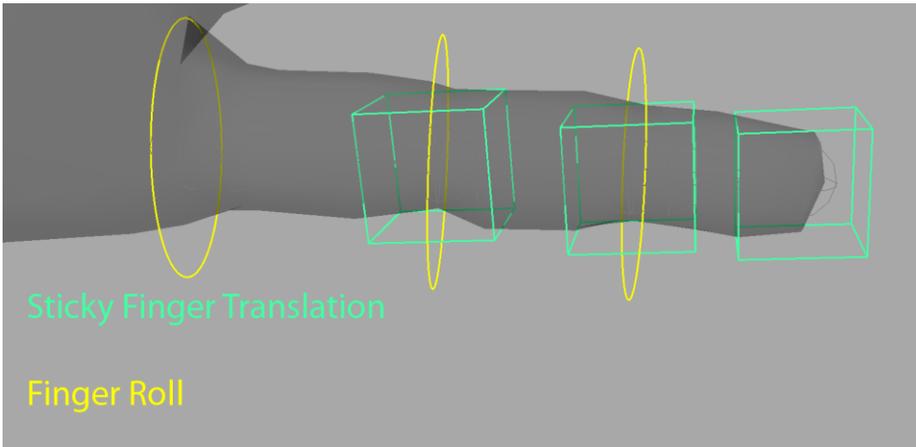
The non-pivot fingers all have hierarchies of segment rotation controllers from the base to the tip segment by default as in Figure 3.49 and 3.50 with each segment having the ability to be made Sticky as in Figure 3.51 to 3.53. Finger stickiness propagates up the finger from the tip to the first joint from the base with translation controllers defining the point in space to which the sticky segments will try to stick and individual roll controllers on each sticky segment as shown in Figure 3.54 to 3.56.



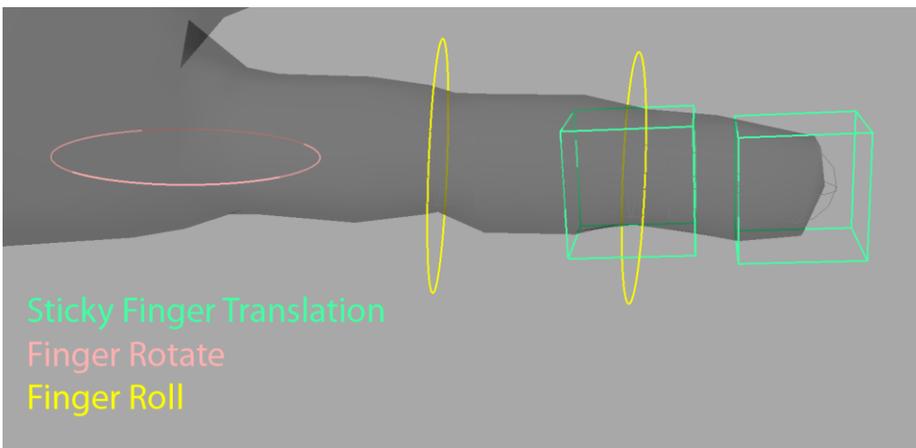
**Figure 3.49 Controllers of the non-sticky Finger – Pinkie in this case – of the IK/DK Hand.**



**Figure 3.50 Motion of the non-sticky Finger – Index in this case – of the IK/DK Hand.**



**Figure 3.51 Controllers of the Sticky1 Finger – Pinkie in this case – of the IK/DK Hand.**



**Figure 3.52 Controllers of the Sticky2 Finger – Pinkie in this case – of the IK/DK Hand.**

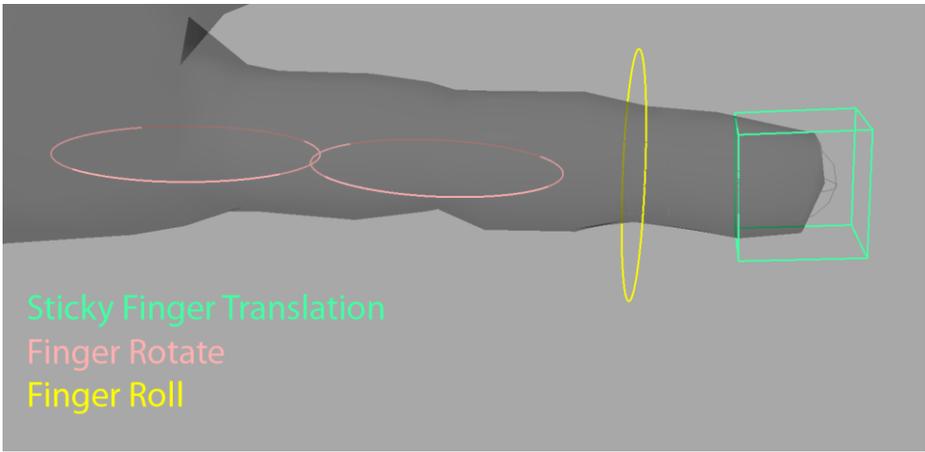


Figure 3.53 Controllers of the Sticky3 Finger – Pinkie in this case – of the IK/DK Hand.

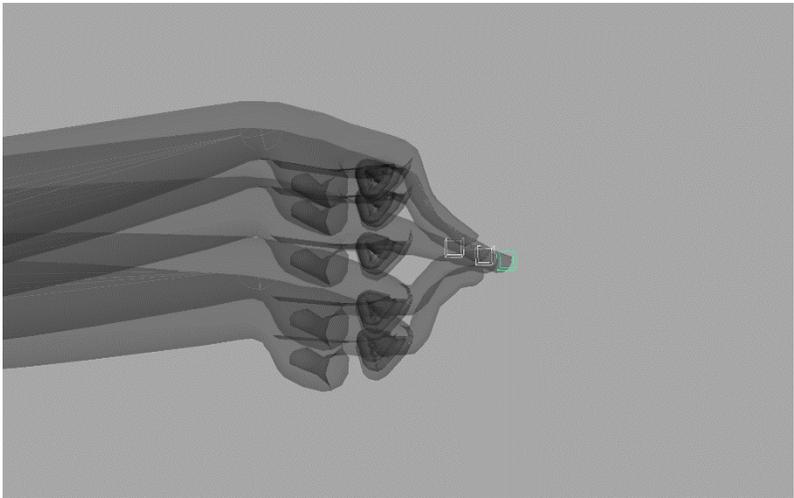


Figure 3.54 Motion of the Sticky1 Finger – Index Sticky1 in this case – of the IK/DK Hand.

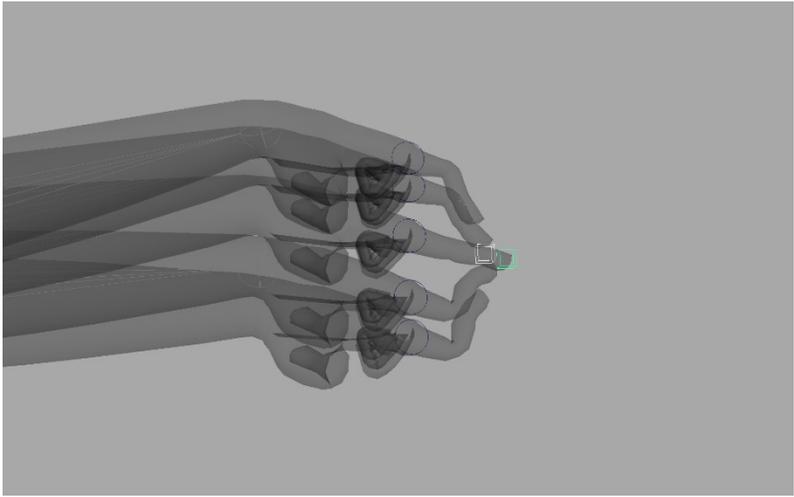
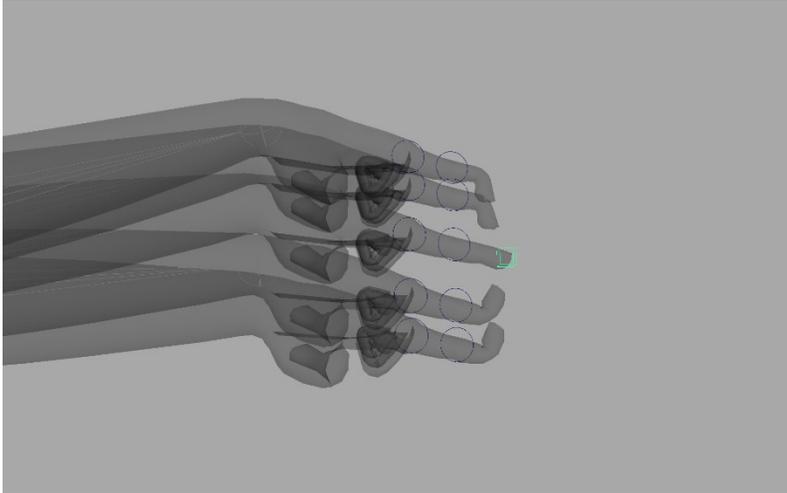


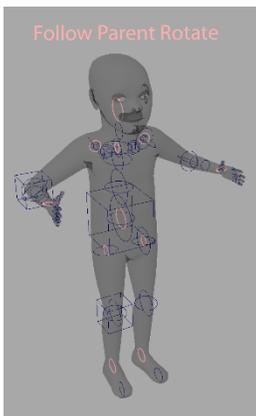
Figure 3.55 Motion of the Sticky2 Finger – Index Sticky2 in this case – of the IK/DK Hand.



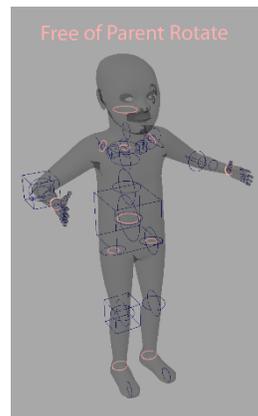
**Figure 3.56 Motion of the Sticky3 Finger – Index Sticky3 in this case – of the IK/DK Hand.**

### **3.7 Follow/Free Connectors**

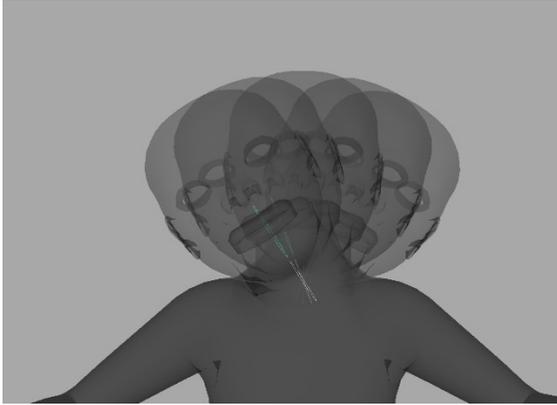
During animation, it is sometimes better if some connected body parts don't follow the rotation of the part they are attached to, so the MDP rig has Follow/Free connectors that can be toggled between following the orientation of their parent in the hierarchy or not as shown in Figure 3.57 and 3.58. These are on the head-neck, neck-spine, pelvis-spine, shoulder-clavicle, hip-pelvis, FK/MK wrist-lower arm and FK/MK ankle-lower leg connections as in Figure 3.59 to 3.71.



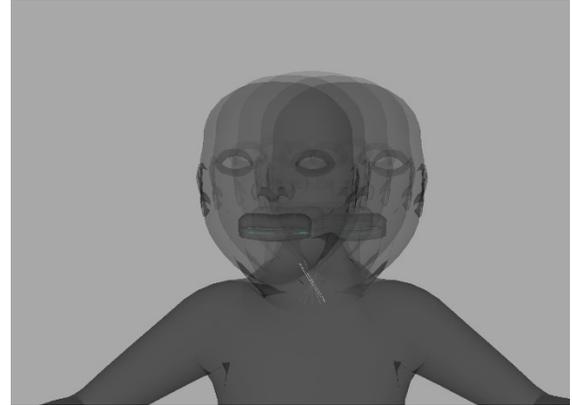
**Figure 3.57 All the Follow-Parent rotation controllers of the whole body.**



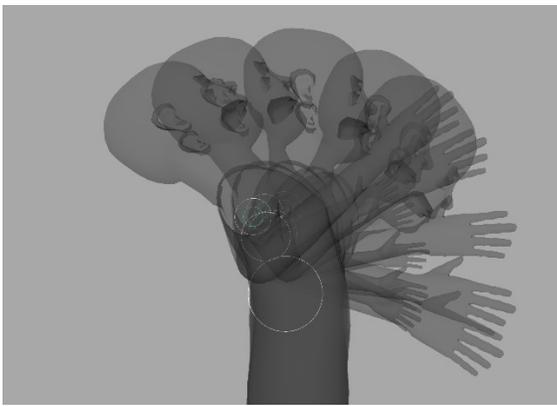
**Figure 3.58 All the Free-of-Parent rotation controllers of the whole body.**



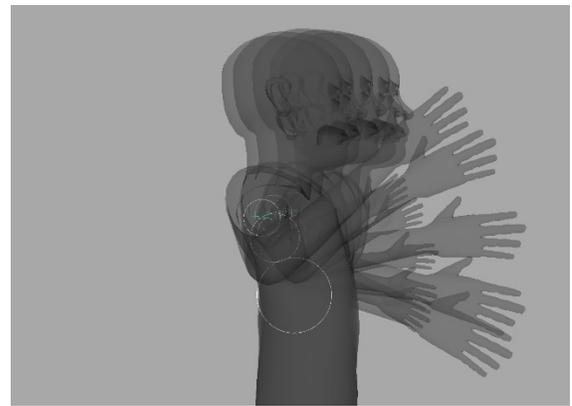
**Figure 3.59 Motion of the Follow-Parent Head.**



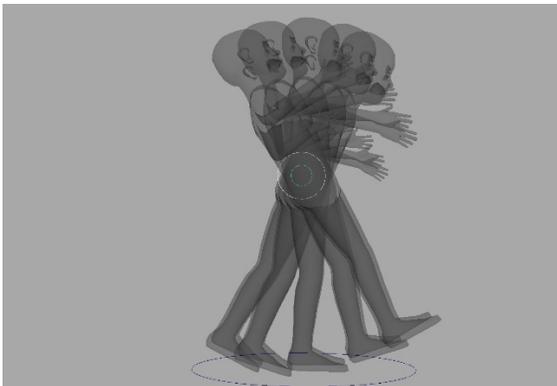
**Figure 3.60 Motion of the Free-of-Parent Head.**



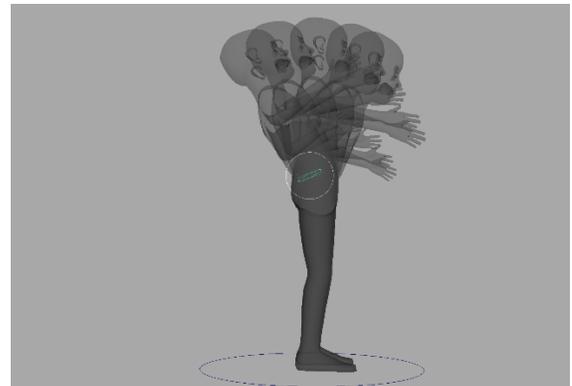
**Figure 3.61 Motion of the Follow-Parent Spine-Neck connector.**



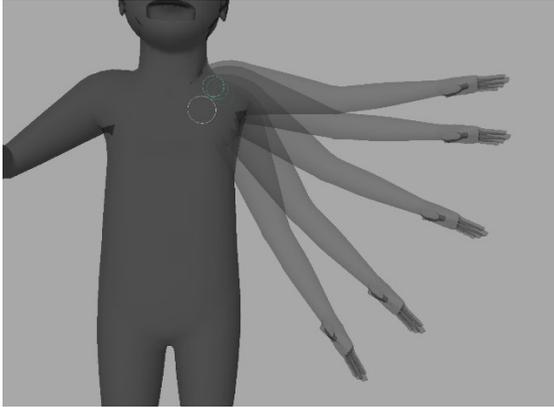
**Figure 3.62 Motion of the Free-of-Parent Spine-Neck connector.**



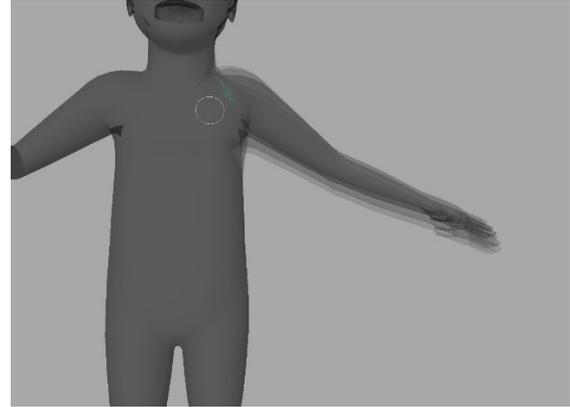
**Figure 3.63 Motion of the Follow-Parent Spine-Pelvis connector.**



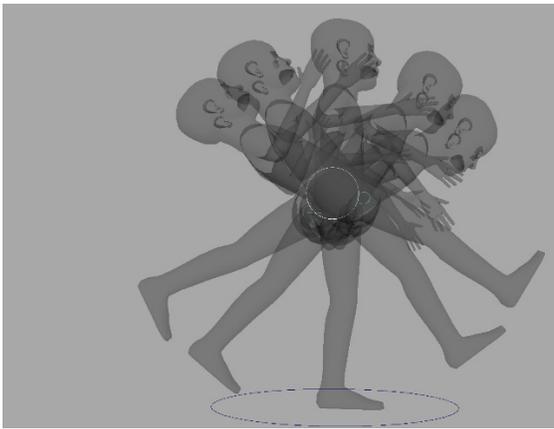
**Figure 3.64 Motion of the Free-of-Parent Spine-Pelvis connector.**



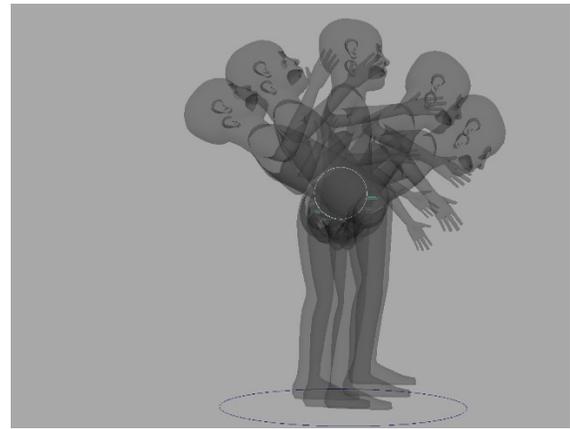
**Figure 3.65 Motion of the Follow-Parent Arm connector.**



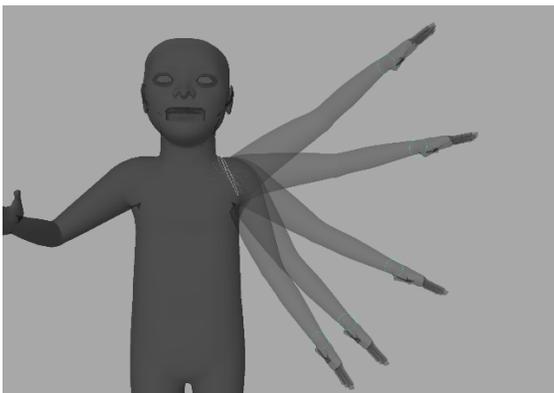
**Figure 3.66 Motion of the Free-of-Parent Arm connector.**



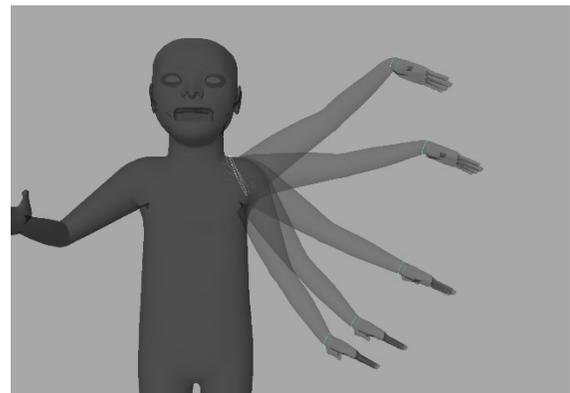
**Figure 3.67 Motion of the Follow-Parent Leg connector.**



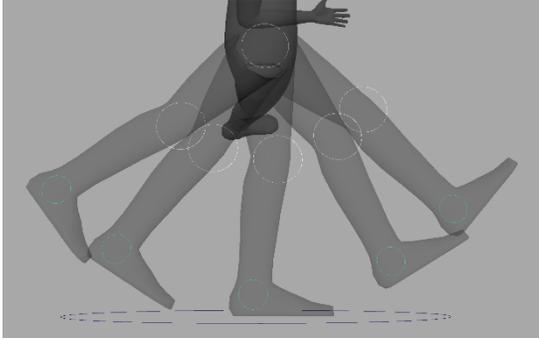
**Figure 3.68 Motion of the Free-of-Parent Leg connector.**



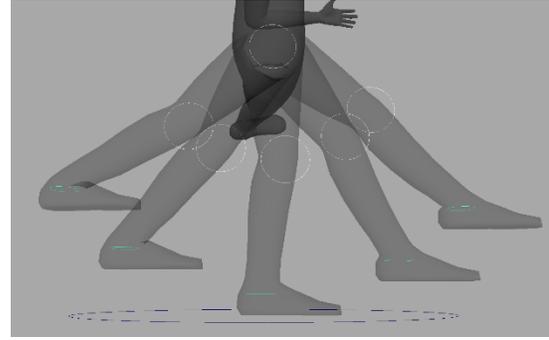
**Figure 3.69 Motion of the Follow-Parent FK/MK Wrist connector.**



**Figure 3.70 Motion of the Free-of-Parent FK/MK Wrist connector.**



**Figure 3.71 Motion of the Follow-Parent FK/MK Ankle connector.**



**Figure 3.72 Motion of the Free-of-Parent FK/MK Ankle connector.**

## Chapter 4: MDP Rig Implementation

The MDP rig has 3 levels of abstraction between the geometry mesh that's being deformed and the controllers enabling the animation:

1. Skin-joints: joints to which the mesh is attached via "skinning" – these are all the same-level children of a root node that never moves from origin as to avoid the possibility of double transformation.
2. Align-nodes: pure transformation node having nothing but position-rotation-scale values to which the skin-joints are position-rotation-scale constrained – these have the same flat hierarchy on their own root node as the Skin-joints for the same double translation avoidance issues.
3. Rig-nodes: the nodes that structure the controllers to which the Align-nodes are position-rotation-scale constrained – they are divided by body part with each part having its collection of mutually exclusive master pivot sub-rigs into which the rig-nodes are organized.

The MDP rig implementation has 2 parts:

- The autorig code that constructs and organizes all the controllers, attributes and helper nodes that makeup the rig.
- The user interface button panels that run the auto-aligning, keying, deletion and selection scripts on demand.

The MDP rig was implemented in Autodesk Maya 2019 using Python, which is application independent. Universal 3D mechanisms such FK, IK expressions and constraints were employed as the connective tissue of the rig. All trigonometric and geometric calculations

were coded in Python. No Maya-specific features were used, thus ensuring that the MDP rig design could be implemented in any other 3D software package, such as Blender.

#### **4.1 Sub-Rig Switching**

The mode switching on which the MDP rig relies is accomplished by position-rotation-scale constraining the align-nodes to their body part's sub-rigs then switching which sub-rig constraint is active. One and only one sub-rig is active at once and, likewise, so is one and only one constraint. Align-nodes are never unconstrained by sub-rig, as this may cause unpredictable transformations that could break the rig. Similarly, there is no offsets in the position-rotation-scale values between the align-nodes and their constraining rig-nodes due to the possibility of unintended transformation.

#### **4.2 Word Alignment**

The smooth transition between sub-rigs is based on the alignment of the world-coordinate position and rotation of the switched to sub-rig to its corresponding align-nodes. In this way, when the sub-rig becomes active, the align-nodes won't actually change transformation even as they are constrained by different sub-rig then they were. FK Rotate controllers align their world rotation to the world rotation of their align-nodes and IK Translation controllers align their world position to the world position of their align-nodes. Scale alignment is usually done by matching the scale of the FK Rotate controllers to their align-nodes since scaling is done along the rotated axes. Lift controllers that affect the orbit of child controllers about the centre of an align node without rotating the node itself align to the rotation of helper nodes that are positioned constrained to the centre node and aim constrained to always point one of their axes at the child node.

### 4.3 Aligning IK

Aligning the double segment IK chains of the limbs requires more calculation than FK Rotation, single-segment IK end Translation or Lift orientation. The end-joint of the IK chain can easily be world position aligned but after that the IK Rotate controller needs the angle between where the IK mid-joint currently is and corresponding where the align-node is. This is calculated by:

1. Calculate the normal vector of the plane defined by the IK top-, mid- and end-joints using the cross product of the IK mid-top vector and IK mid-end vector.
2. Calculate the normal vector of the plane defined by the align-node top-, mid- and end-points using the cross product of the align mid-top vector and IK mid-end vector.
3. Get the angle between the IK normal vector and the align normal vector.

Thus, it is determined the angle by which to turn the IK Rotate controller to match the position of the IK mid-joint to that of the align-node limb mid-point.

## Chapter 5: User Interface of the MDP Rig

The MDP rig's user interface is composed of two windows – switchUI, as in Figure 5.1, for switching between the modes of each body part and selKeyUI, as in Figure 5.27, for selecting and keying the controllers of each body part. Both UI windows are composed of buttons and drop-down menus laid out in a pattern resembling a sprawled humanoid form – head, neck, spine and pelvis down the centre with arm, hands and fingers on the middle left and right and the legs on the lower left and right. This body-shaped layout is reminiscent to those used by controller picker UIs and was used in order to lower the animators' decision-to-selection time (Hick, 1952) (Hyman, 1953) based on familiarity with which buttons correspond to which body part, due to their placement relative to each other.

The animation workflow will make greater use of the switchUI window as the motion of the body and which pivots should lead that motion during which intervals are blocked out. The selKeyUI will increase in use as the animation is refined and more keying of controllers is done between mode switch keyframes and increase even further as active sub-rig controllers are selected to have their keyframes refined in the later stages of animation.

The switchUI window also has visibility checkboxes for the pelvis and hip controllers which have no alternative pivot modes to switch between. They can be necessary in the posing of the lower body but also have large controllers that can get in the way when not needed hence the ability to hide them. Similarly, the selKeyUI has visibility checkboxes for each limb's roll controller, which are used for countering geometry candywrapping and deemed to be more useful during the animation refinement stage. These roll controllers are

located on each limb's mid-joint and can get in the way of the manipulation of the more often used lower limb controller. Therefore, they are easily hideable in the selKeyUI window, which will get the most use during late-phase animation.

## 5.1 switchUI

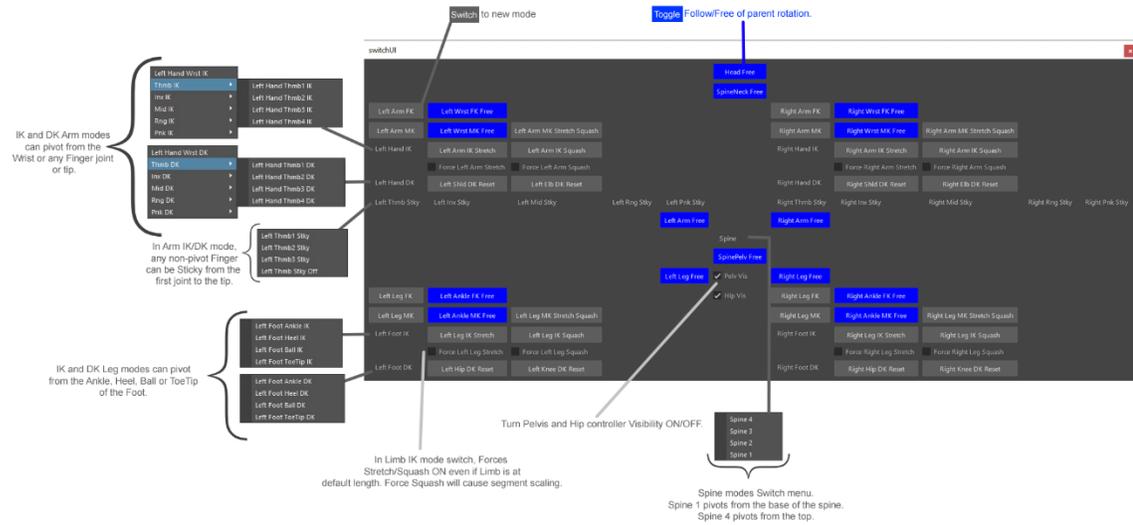
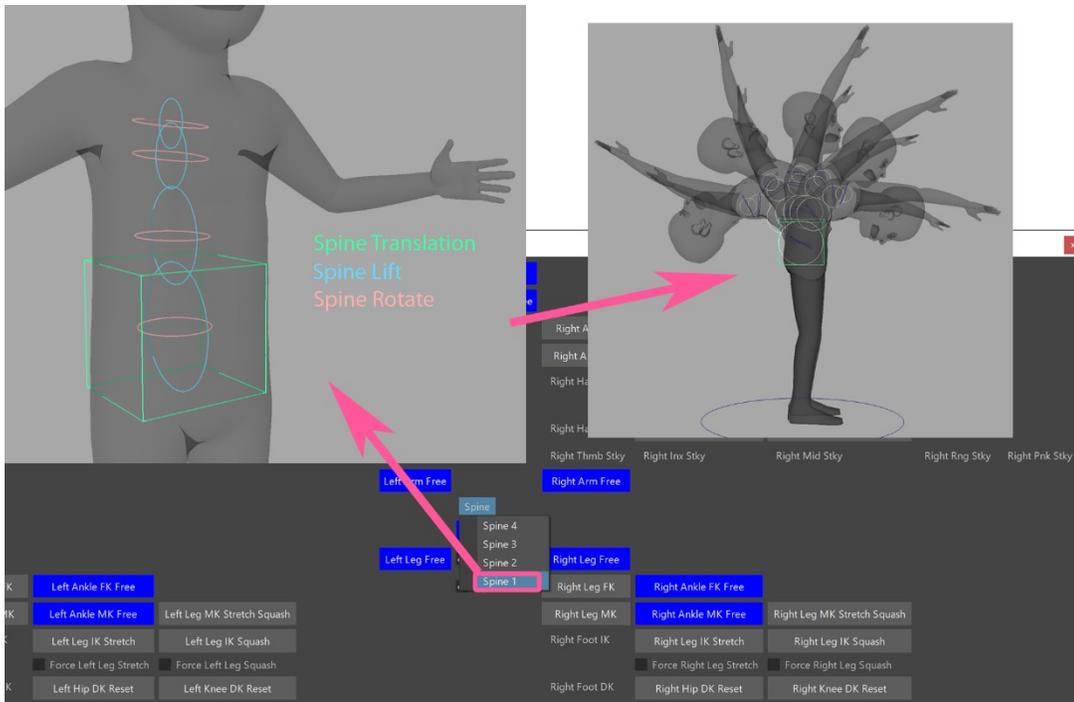


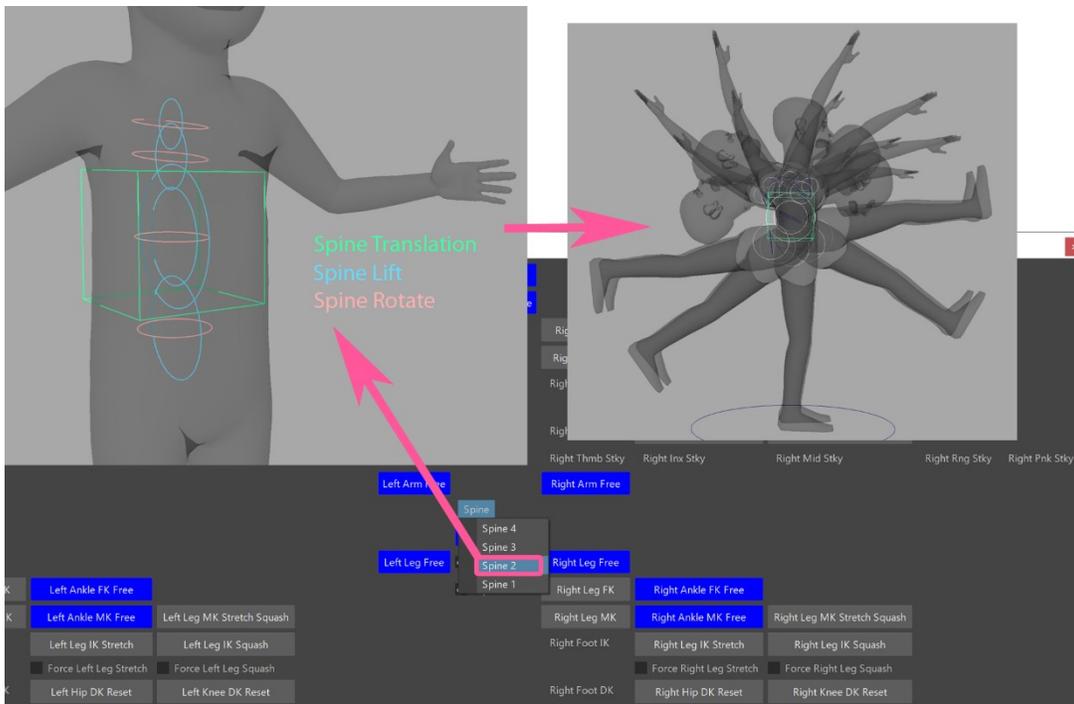
Figure 5.1 The switchUI legend.

### 5.1.1 Spine

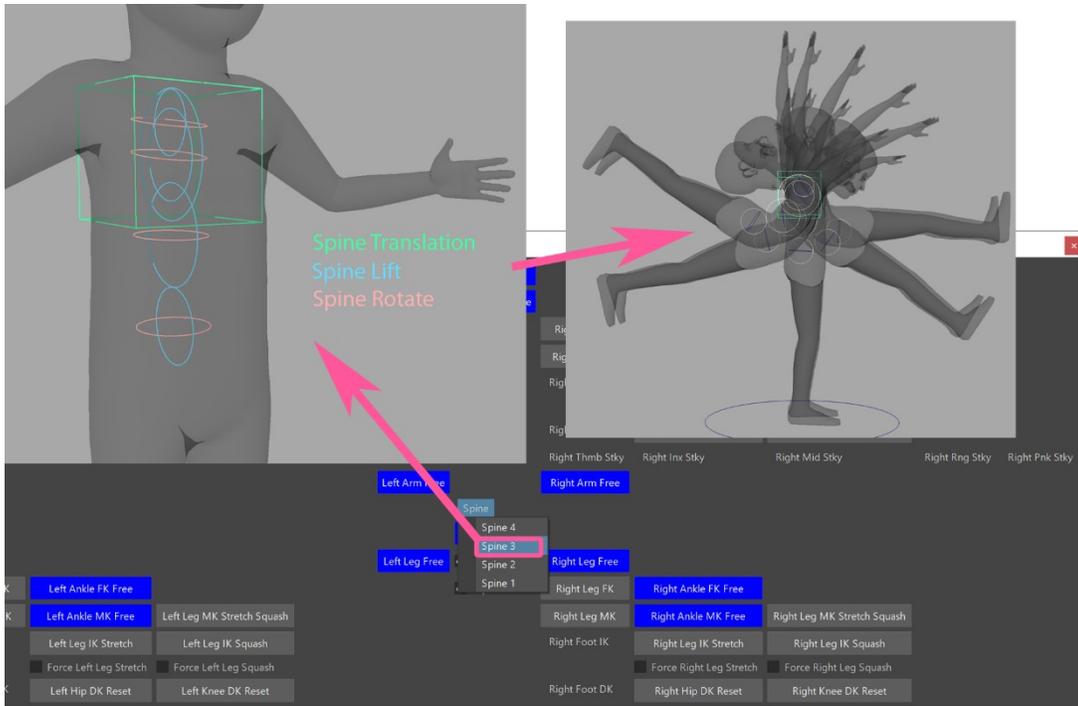
The spine mode switch drop-down menu is laid out in descending order from Spine4 to the Spine1 to match the layout of the MDP rig spine pivot modes with Spine1 at the pelvis and Spine4 the neck as shown in Figure 5.2 to 5.5.



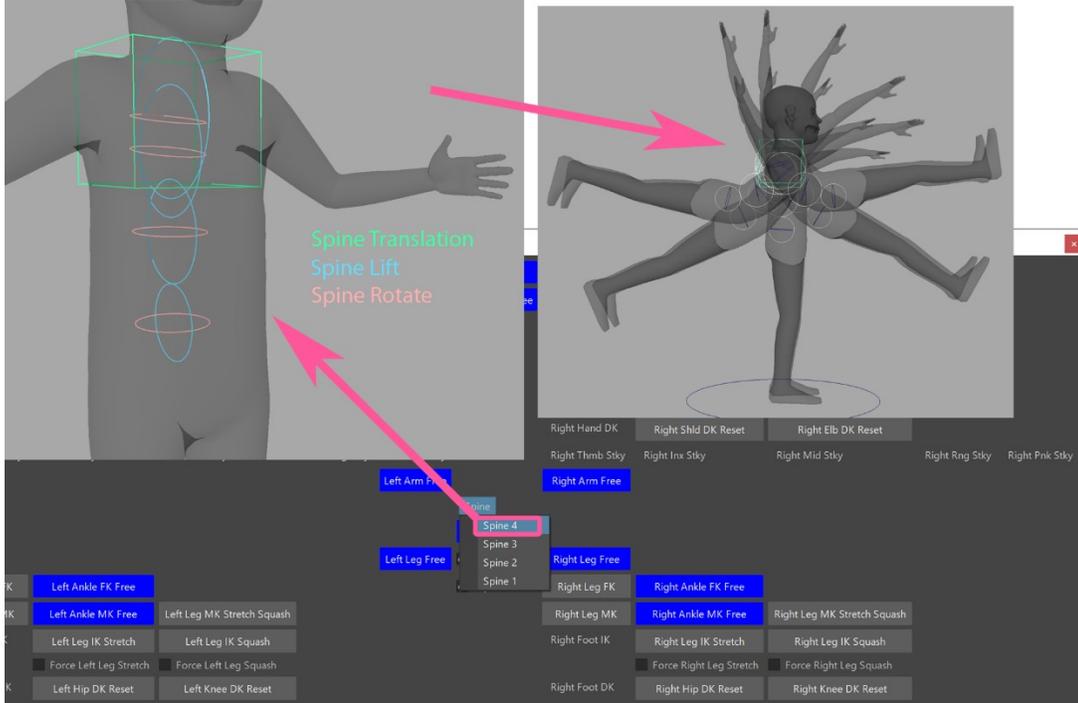
**Figure 5.2 Spine1 button triggers mode switch to Spine1 controllers to create Spine1 pivot-centric motion.**



**Figure 5.3 Spine2 button triggers mode switch to Spine2 controllers to create Spine2 pivot-centric motion.**



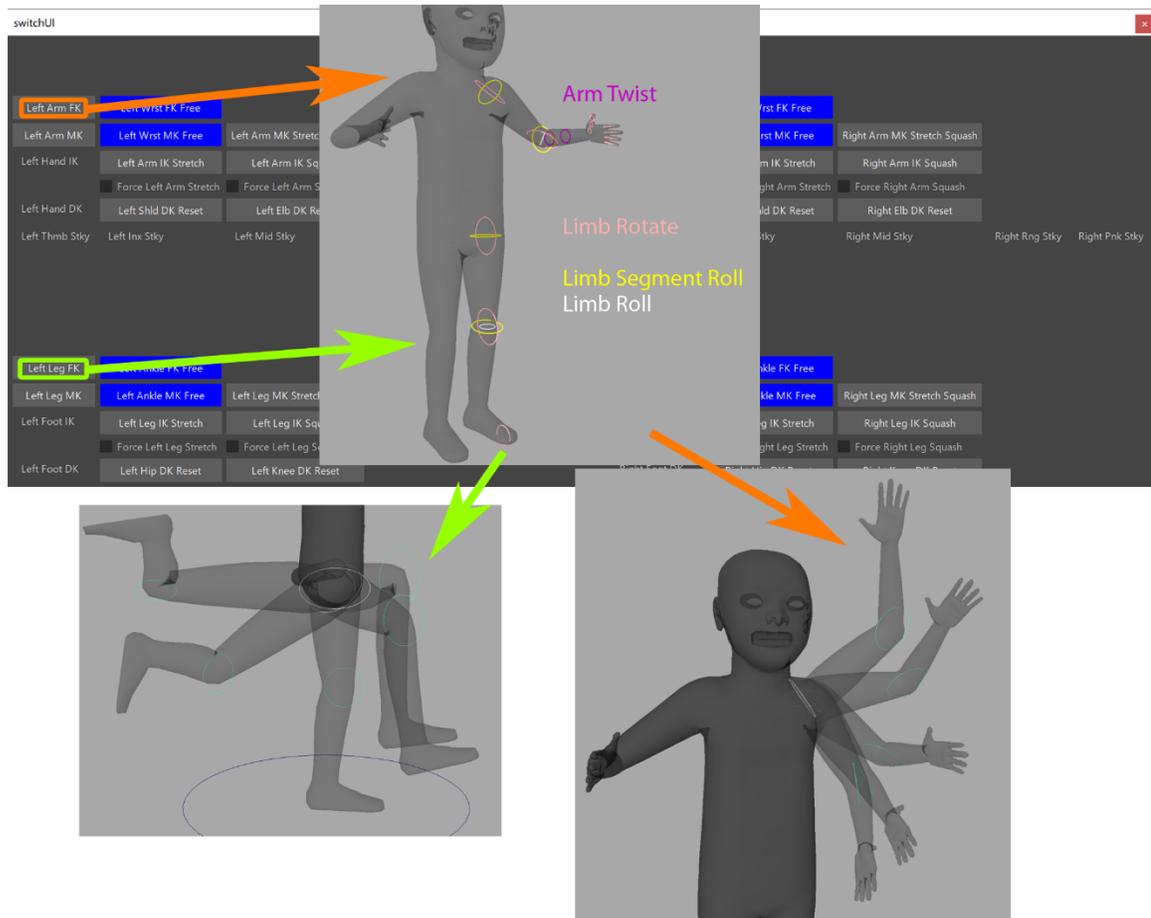
**Figure 5.4 Spine3 button triggers mode switch to Spine3 controllers to create Spine3 pivot-centric motion.**



**Figure 5.5 fig. Spine4 button triggers mode switch to Spine4 controllers to create Spine4 pivot-centric motion.**

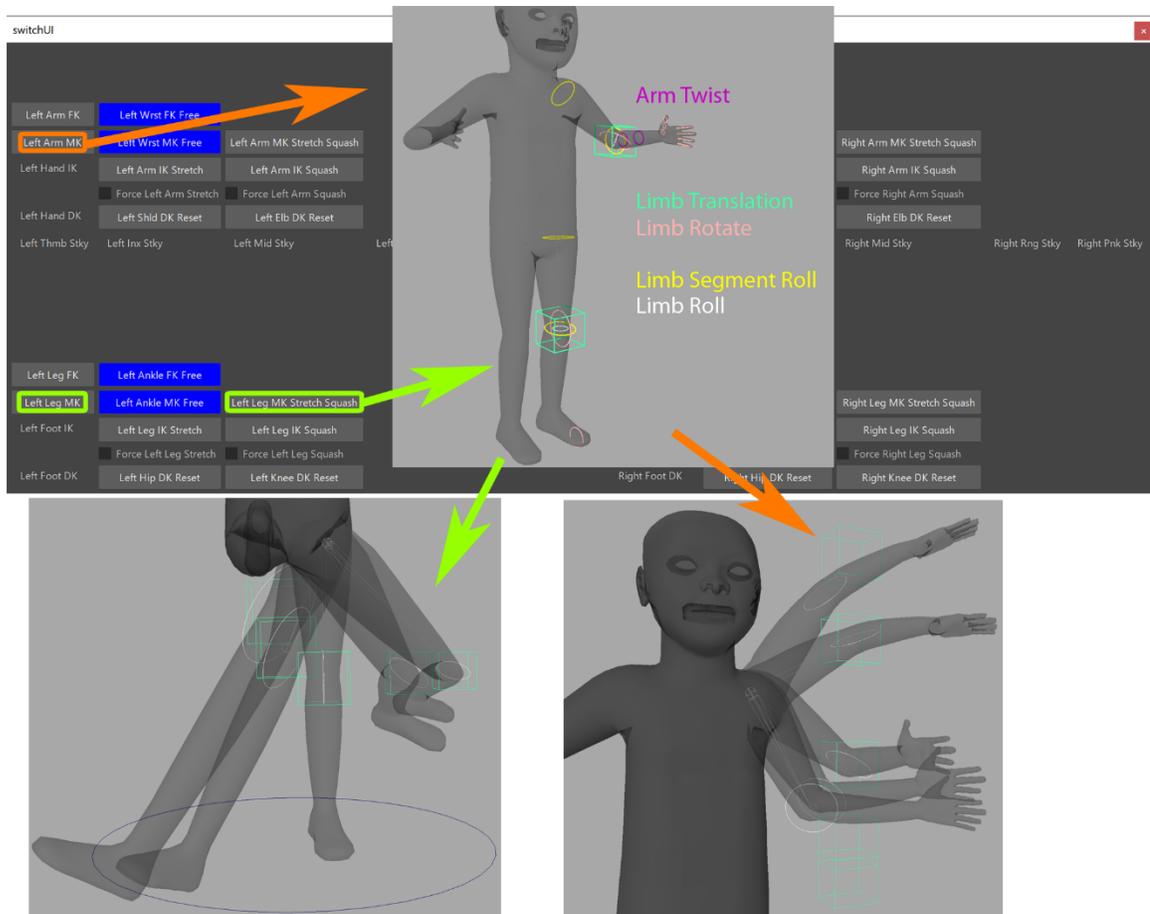
### 5.1.2 Limb

The arm and leg FK mode correspond to just one button, since the sub-rig is a simple hierarchy of rotation-scaling controllers as in Figure 5.6.



**Figure 5.6 Arm/Leg IK button triggers mode switch to Arm/Leg IK controllers to create Arm/Leg IK motion.**

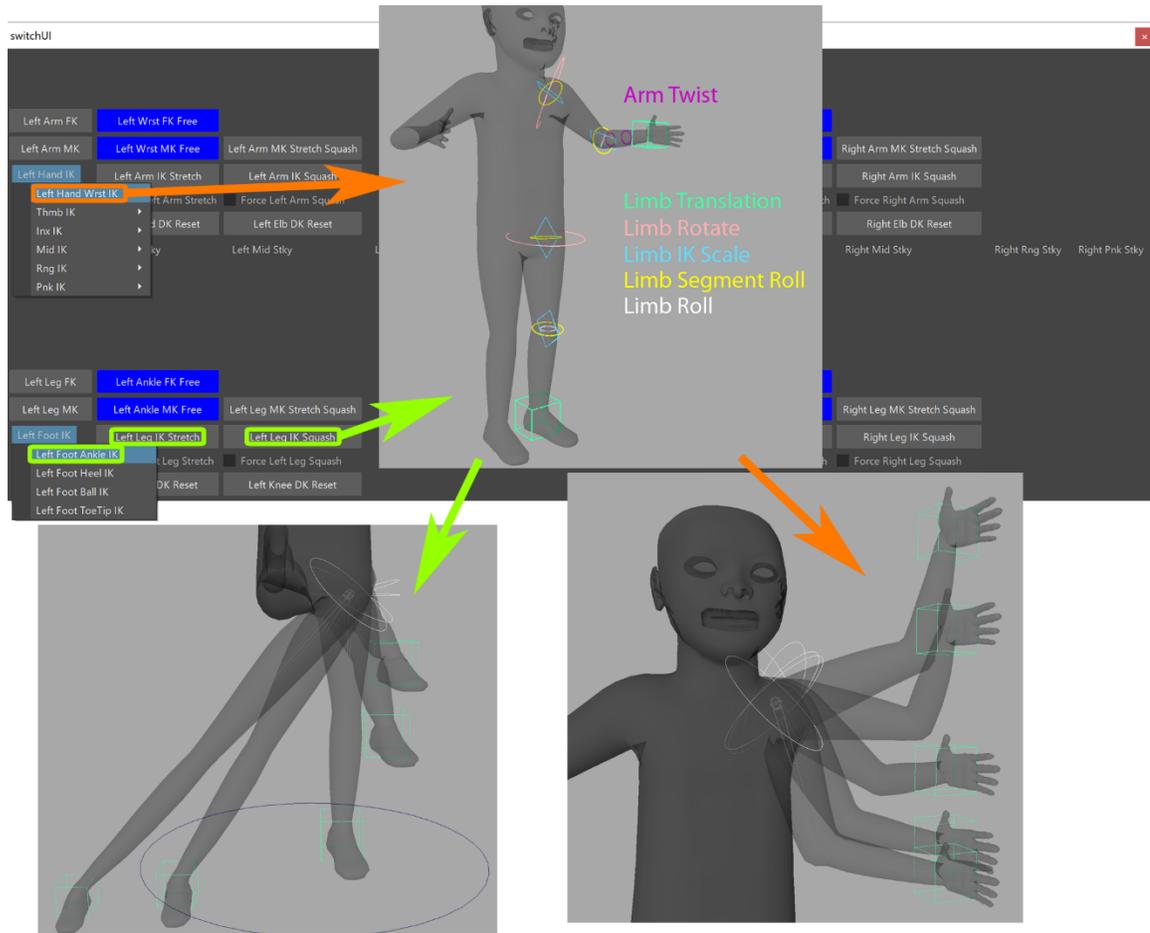
The MK mode has an IK bone in the upper arm/leg, so it has a squash-and-stretch toggle button "MK Stretch Squash" that dictates whether the bone rotates to point at the mid-point translation controller (default squash-and-stretch OFF as shown in the arm) or the bone also scales so that its end-joint– the limb's mid-point – is positioned exactly on the translation controller (squash-and-stretch ON as shown in the leg) as in Figure 5.7.



**Figure 5.7 Arm/Leg MK button triggers mode switch to Arm/Leg MK controllers to create Arm/Leg MK motion.**

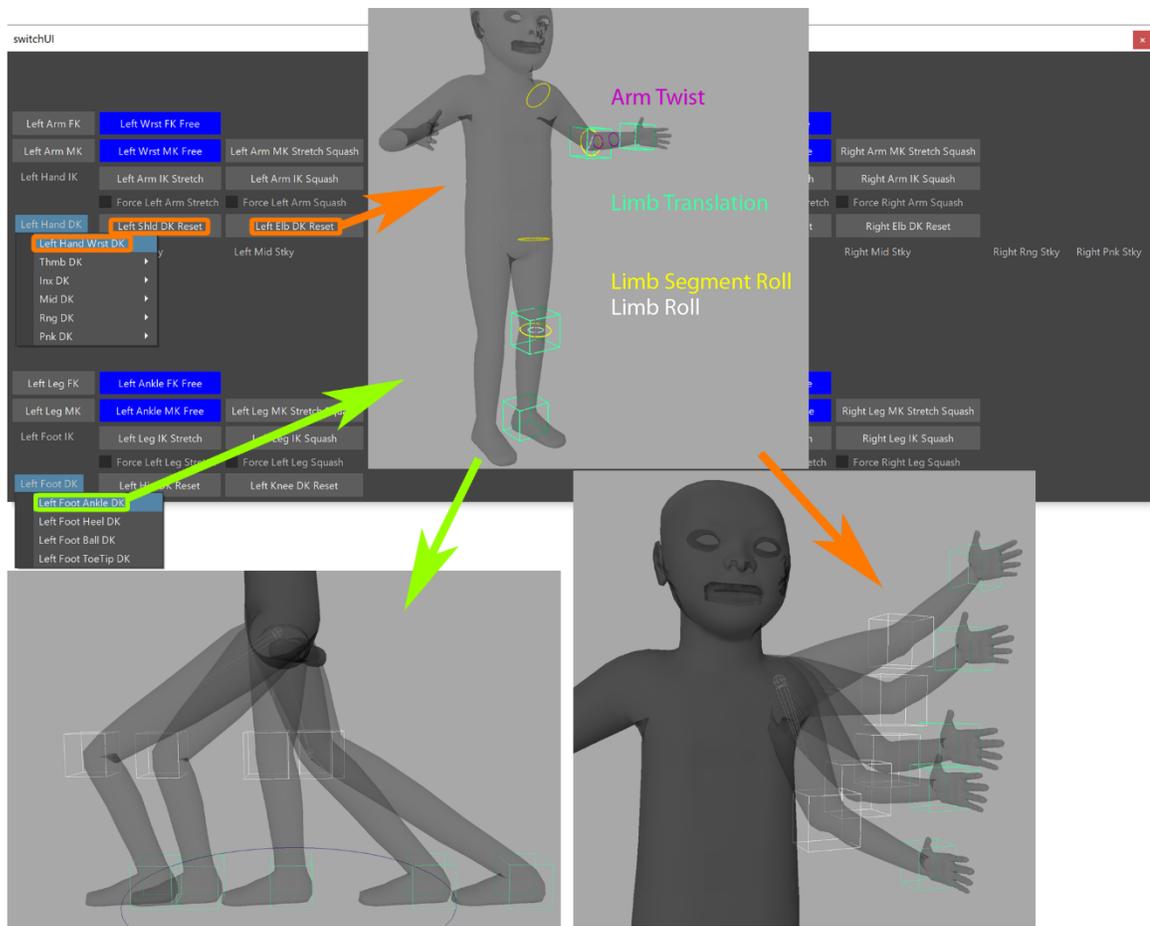
The IK mode is a two-bone IK chain, so it has separate toggle buttons for "IK Stretch" and "IK Squash" which are disabled by default (default squash-and-stretch OFF as shown in the arm) . Stretch enabling causes the limb bones to lengthen so that the end-pivot is always in the centre of the pivot translation controller when the distance between the top-joint and translation controller exceeds that of the original total length of each limb segment. Conversely, squash enabling causes the bones to shorten so that each limb segment is only ever as long as it needs to be for the pivot point to remain centred in the translation controller – stretch-enabled automatically when squash is since stretching can be done without squashing but not vice versa (squash-and-stretch ON as shown in the leg). The

"Force Stretch" and "Force Squash" checkboxes will force the Stretch and Squash attributes to be enabled and the IK limb segment scale controllers to compensate on mode switch to IK even if the limb isn't squashed/stretched for a smooth transition into IK squashed/stretched after the mode switch as in Figure 5.8.



**Figure 5.8 Arm/Leg IK button triggers mode switch to Arm/Leg IK controllers to create Arm/Leg IK motion.**

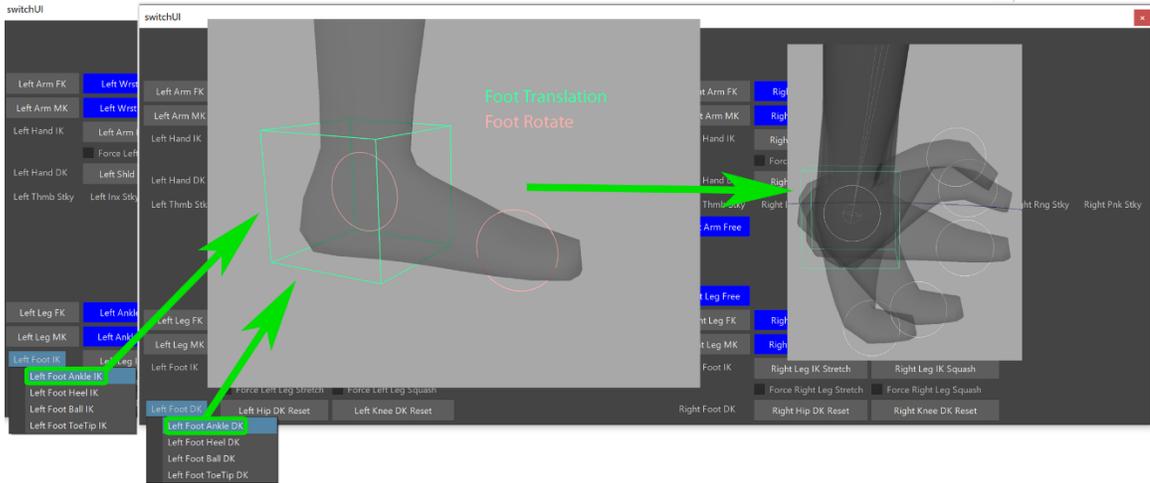
The "Upper DK Reset" and "Lower DK Reset" buttons return the DK limb segments to their original lengths and key their respective Translation controllers in that position (upper and lower DK length reset shown in the arm). The DK limb mode squashes and stretches by its very nature so that the mid-joint and the pivot end-point can be exactly planted in space (upper and lower DK squash-and stretch shown in leg as in Figure 5.9).



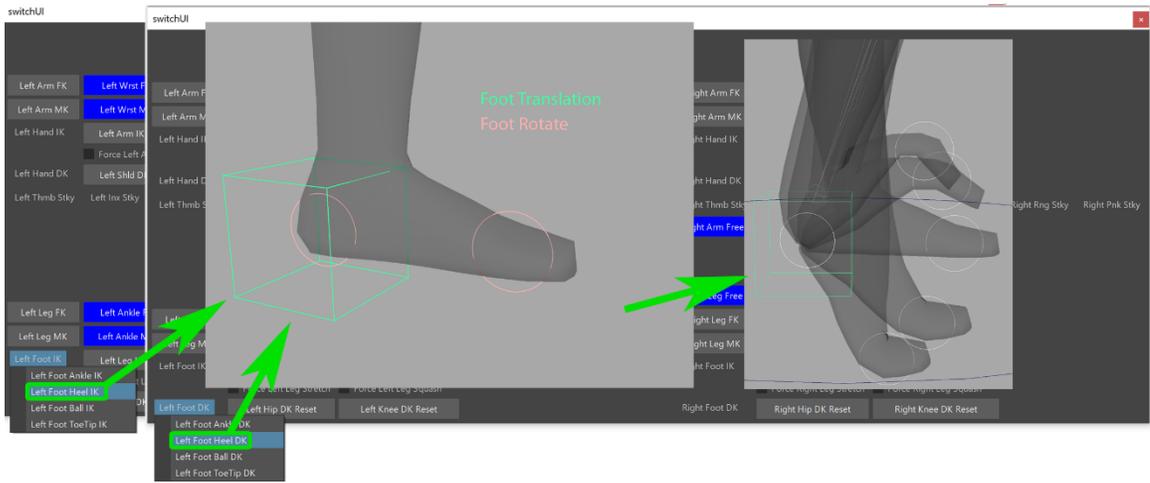
**Figure 5.9 Arm/Leg DK button triggers mode switch to Arm/Leg DK controllers to create Arm/Leg DK motion.**

### 5.1.3 Foot

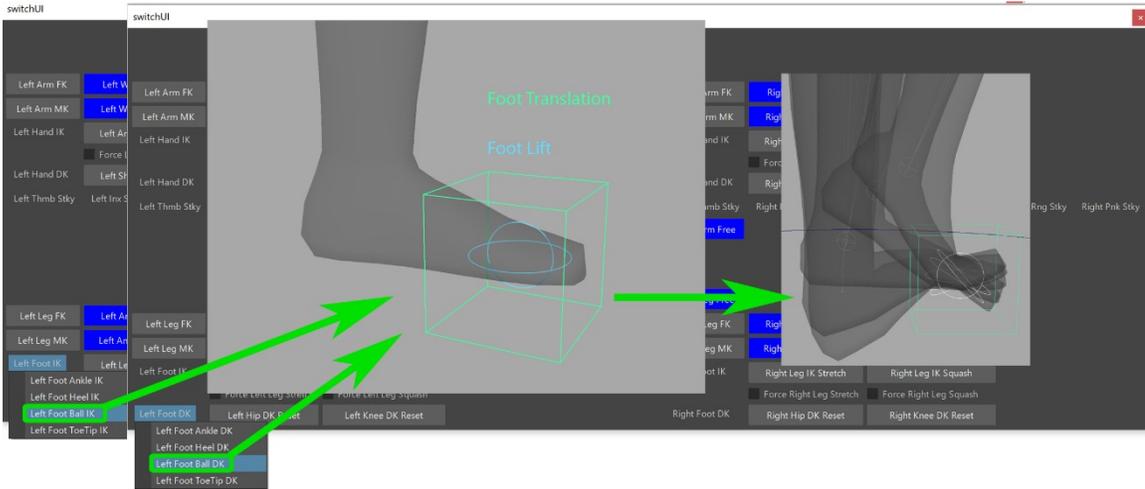
The IK and DK legs have pivot end-points on the ankle, heel, ball and toe tip and their mode dropdown menus are laid out in that order as in Figure 5.10 to 5.13.



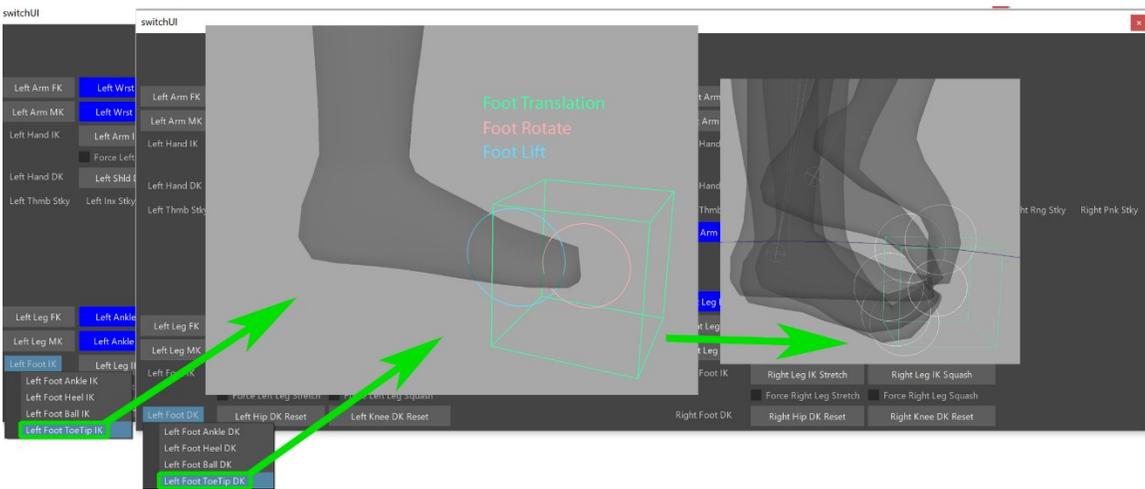
**Figure 5.10 Foot Ankle IK/DK button triggers mode switch to Foot Ankle IK/DK controllers to create Foot Ankle-centric motion.**



**Figure 5.11 Foot Heel IK/DK button triggers mode switch to Foot Heel IK/DK controllers to create Foot Heel-centric motion.**



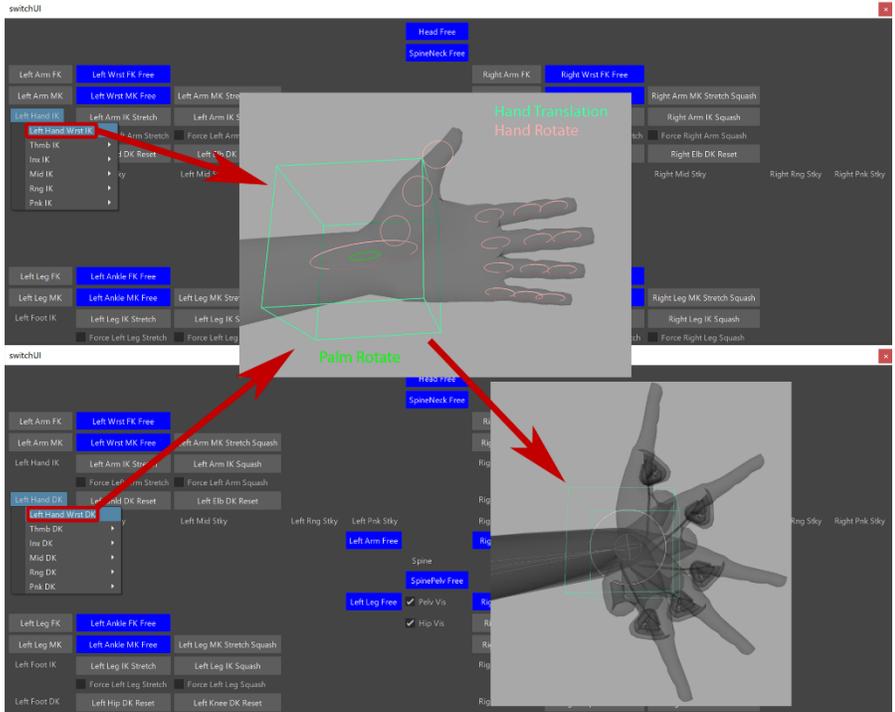
**Figure 5.12 Foot Ball IK/DK button triggers mode switch to Foot Ball IK/DK controllers to create Foot Ball-centric motion.**



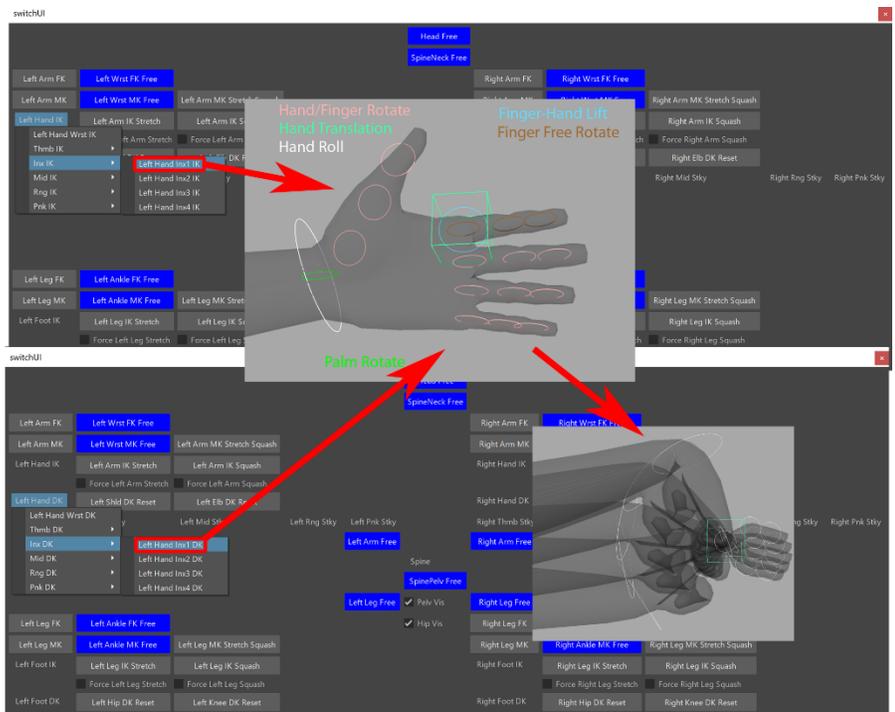
**Figure 5.13 Foot Toe-tip IK/DK button triggers mode switch to Foot Toe-tip IK/DK controllers to create Foot Toe tip-centric motion.**

### 5.1.4 Hand

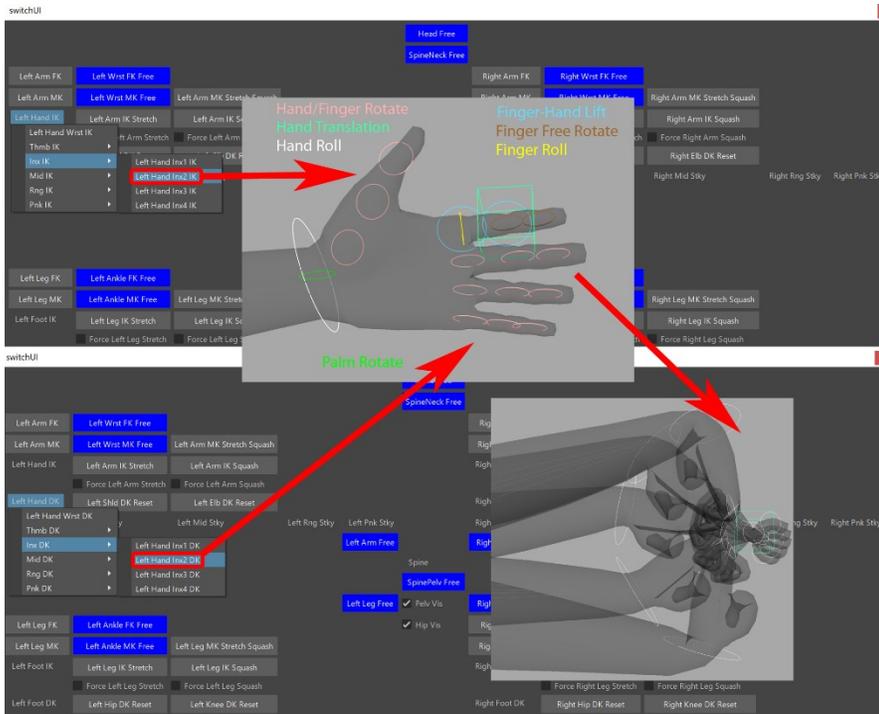
The IK and DK arms have pivot end-points on the wrist and all joints plus tips of each finger so their mode selection dropdown menus are laid out with the wrist on top and sub-menus from thumb to pinkie for each finger's 4 modes as in Figure 5.14 to 5.18.



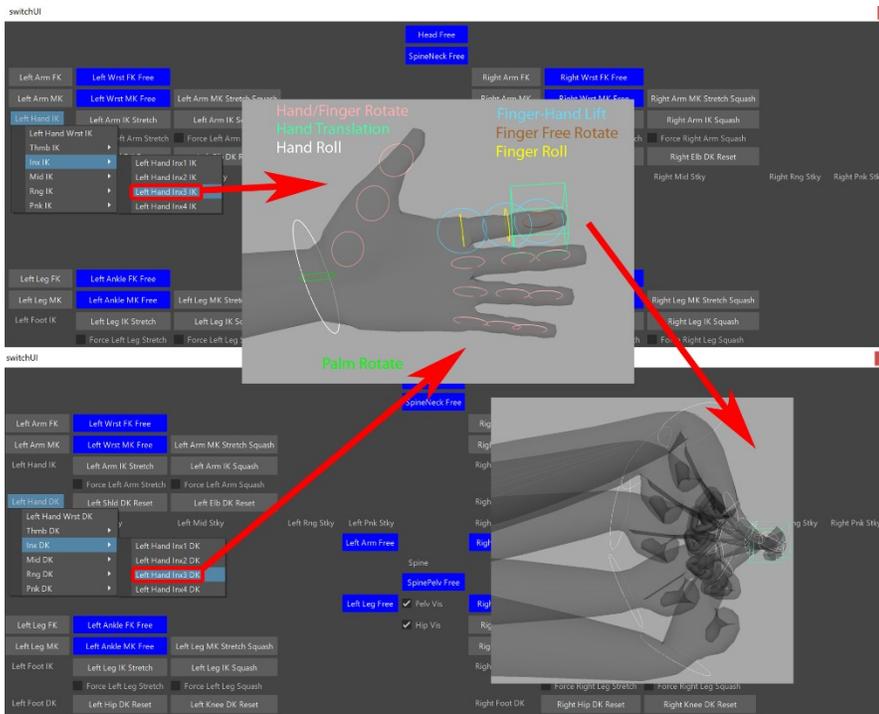
**Figure 5.14 Hand Wrist IK/DK button triggers mode switch to Hand Wrist IK/DK controllers to create Hand Wrist-centric motion.**



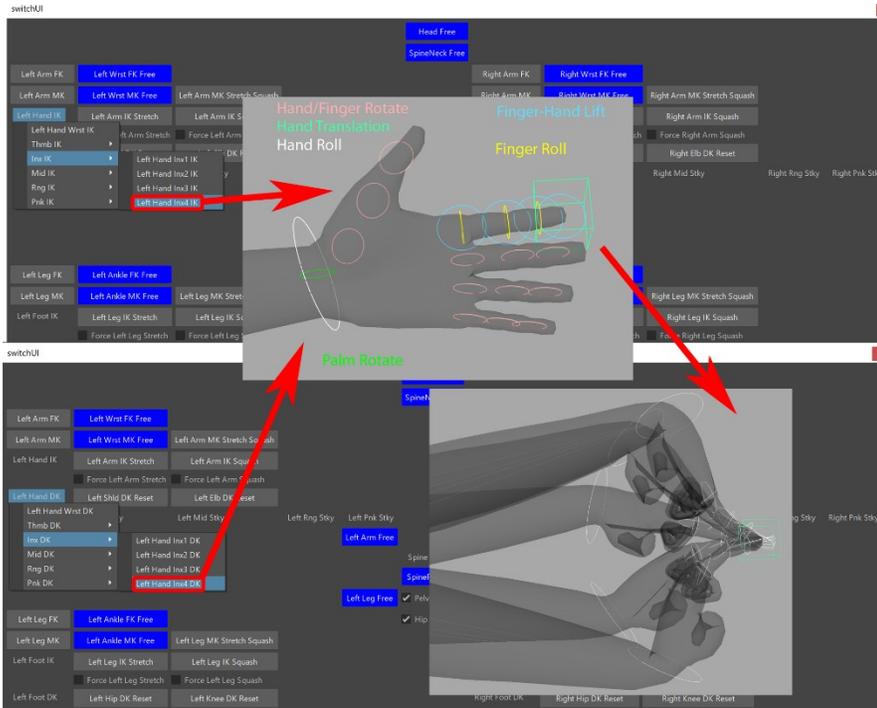
**Figure 5.15 Hand Finger1 IK/DK button triggers mode switch to Hand Finger1 IK/DK controllers to create Hand Finger1-centric motion.**



**Figure 5.16 Hand Finger2 IK/DK button triggers mode switch to Hand Finger2 IK/DK controllers to create Hand Finger2-centric motion.**



**Figure 5.17 Hand Finger3 IK/DK button triggers mode switch to Hand Finger3 IK/DK controllers to create Hand Finger3-centric motion.**



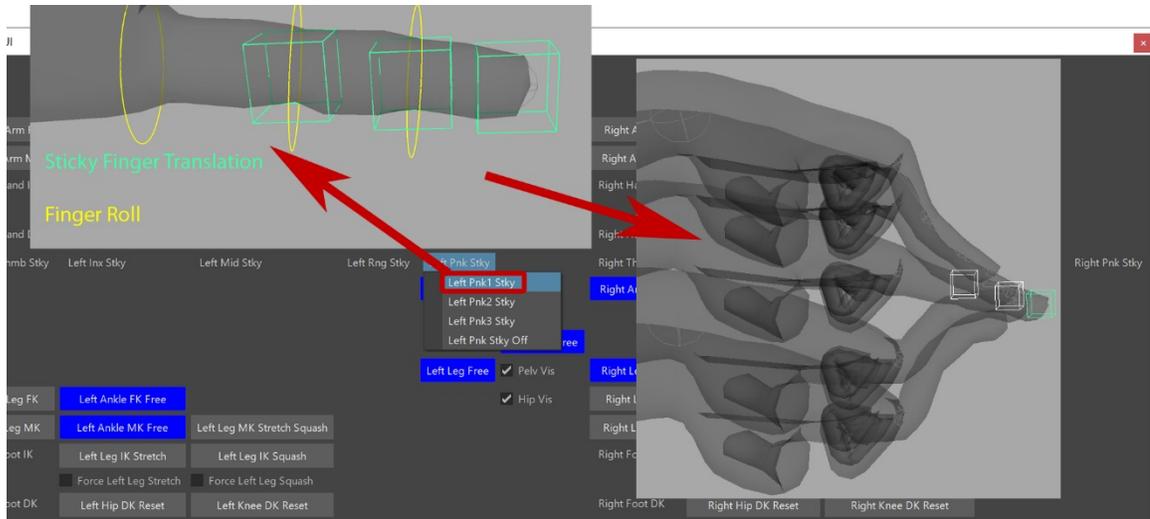
**Figure 5.18 Hand Finger4 IK/DK button triggers mode switch to Hand Finger4 IK/DK controllers to create Hand Finger4-centric motion.**

### 5.1.5 Finger

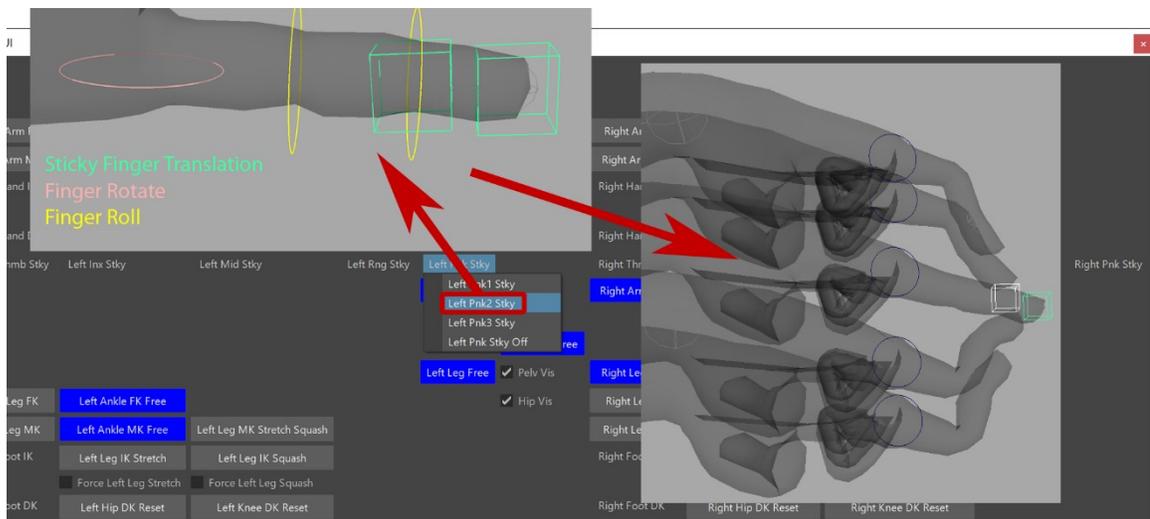
The non-pivot fingers of the IK/DK hand can be made sticky on:

1. "[Finger]1 Stky": the first joint, second joint and tip
2. "[Finger]2 Stky": the second joint and tip
3. "[Finger]3 Stky": the tip

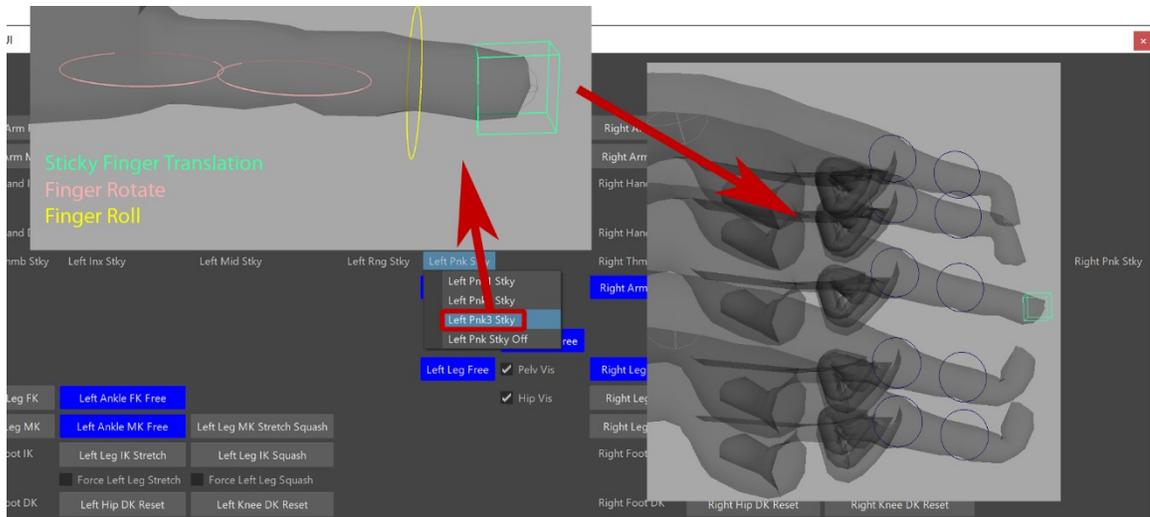
Thus, each finger's sticky dropdown menu are laid out in that order with the final item being the disabling of the stickiness as in Figure 5.19 to 5.22.



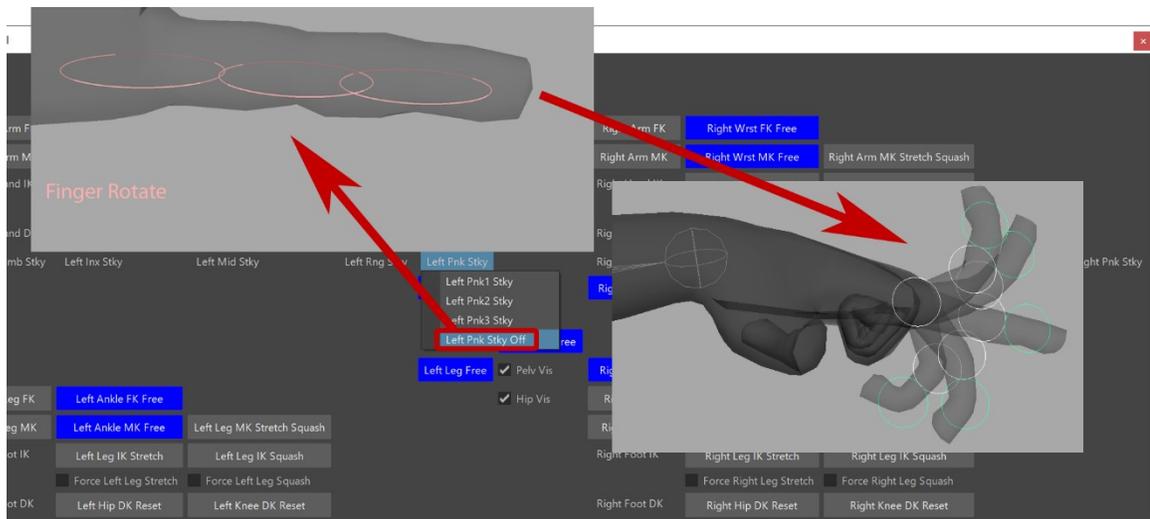
**Figure 5.19 Hand IK/DK Finger Sticky1 button triggers mode switch to Finger Sticky1 controllers to create Finger Sticky1 motion.**



**Figure 5.20 Hand IK/DK Finger Sticky2 button triggers mode switch to Finger Sticky2 controllers to create Finger Sticky2 motion.**



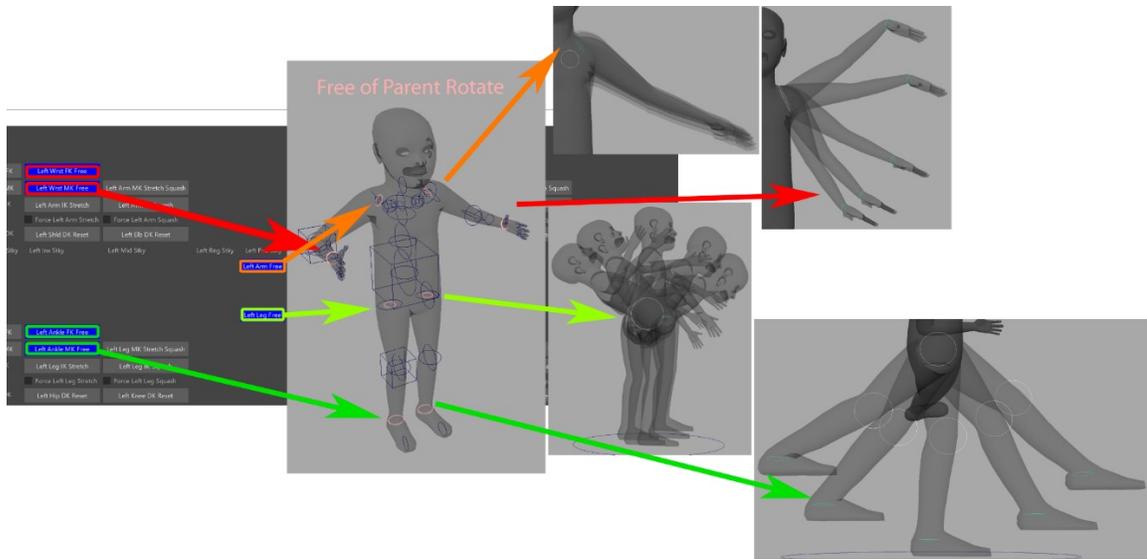
**Figure 5.21 Hand IK/DK Finger Sticky3 button triggers mode switch to Finger Sticky3 controllers to create Finger Sticky3 motion.**



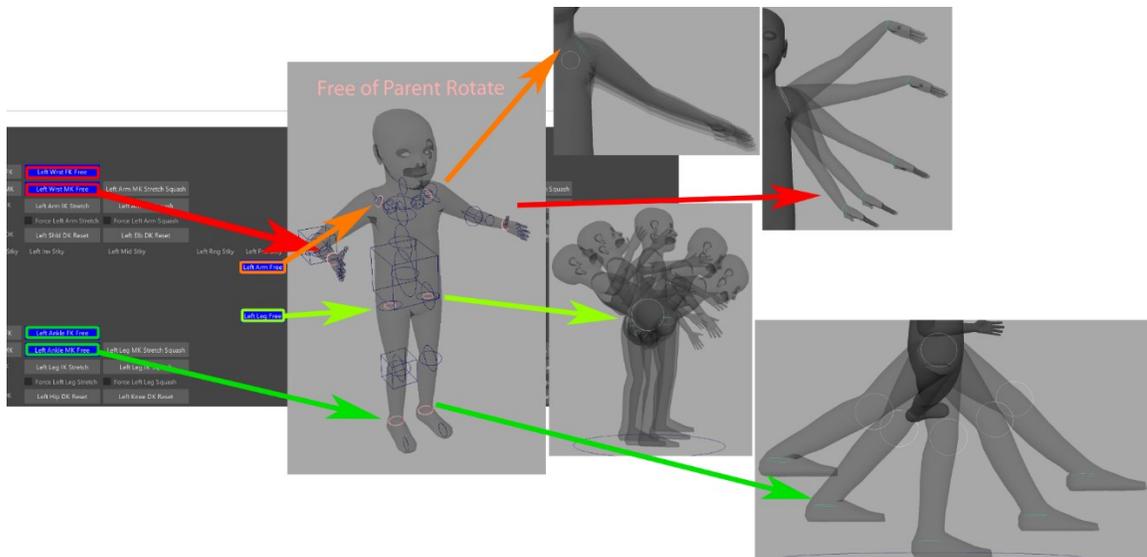
**Figure 5.22 Hand IK/DK Finger Sticky Off button triggers mode switch to default controllers to create Finger Sticky1 motion.**

### 5.1.6 Follow/Free

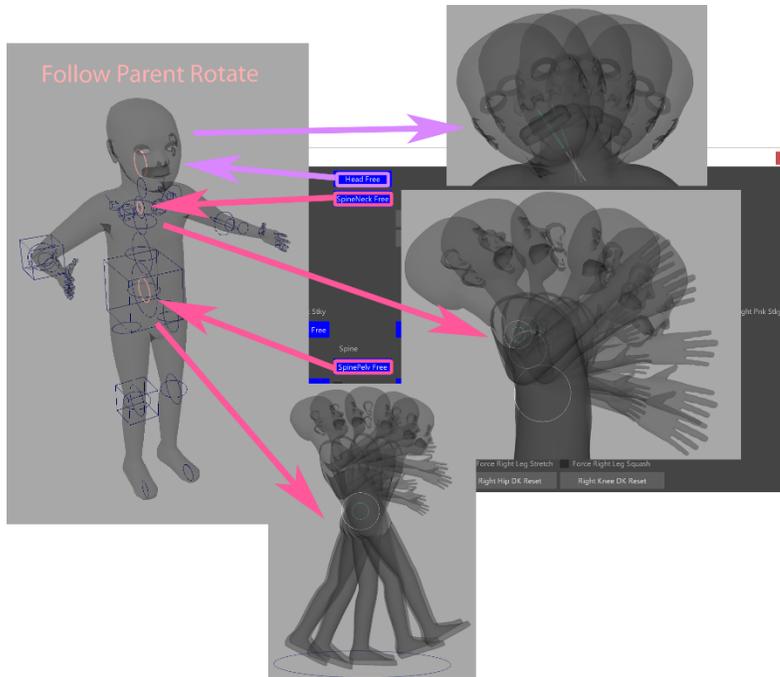
The free/follow connections of the head, neck, pelvis, limbs and FK and MK wrists and ankles are toggled by the blue buttons with the follow mode as the default as in Figure 5.23 to 5.26.



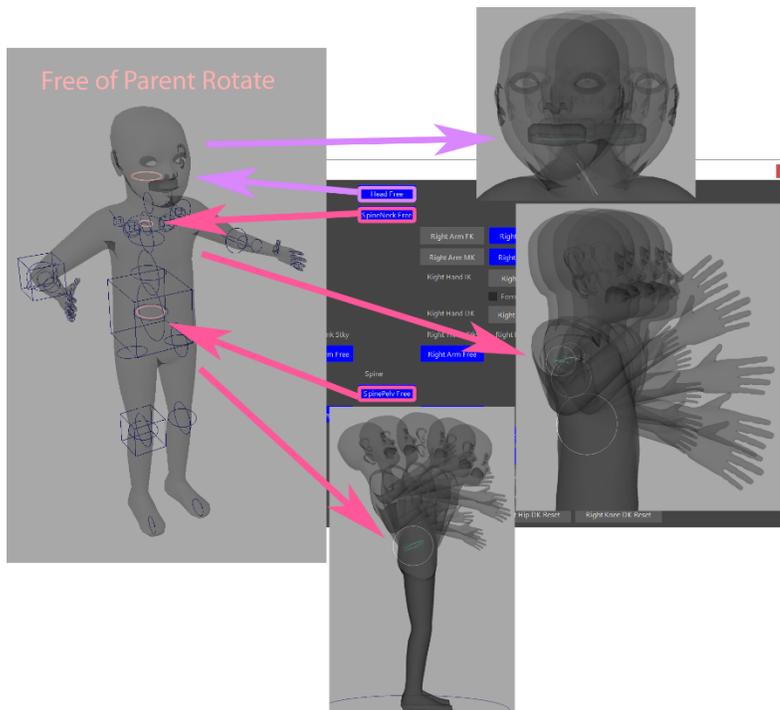
**Figure 5.23 Follow/Free Rotation toggled triggers mode switch to Free-of-Parent Rotate controller if in Follow mode to create motion free of the rotation of the parent controller.**



**Figure 5.24 Follow/Free Rotation toggled triggers mode switch to Free-of-Parent Rotate controller if in Follow mode to create motion free of the rotation of the parent controller.**



**Figure 5.25 Follow/Free Rotation toggled triggers mode switch to Follow-Parent Rotate controller if in Free mode to create motion following the rotation of the parent controller.**



**Figure 5.26 Follow/Free Rotation toggled triggers mode switch to Free-of-Parent Rotate controller if in Follow mode to create motion free of the rotation of the parent controller.**



- "Refresh IK" executes a small time scrub causing IK that's not in the fully evaluated pose to right itself.

The special selection buttons in the upper left area:

- "Sel Attrs" selects all the mode switching attributes of the MDP rig – used to see and edit the keys on which mode switching occur.
- "Sel All" selects all the MDP rig controllers active and inactive – used to manipulate rig posing in the time-line, dopesheet and graph editor.

### **5.2.2 Selection**

All body part selection dropdown menus have the option to:

- "SelAct" selects all the body part controllers of the active mode on the current frame.
- "SelPrev" automatically skips back to the previous mode switch keyframe and selects all the body part controllers of the previous mode.
- "SelAll" selects all the controllers of all the modes of the body part.

### **5.2.3 Keying**

All body part keying dropdown menus have the option to:

- "KeyAct" keys all the body part controllers of the active mode on the current frame at their current keyable attribute values.
- "KeyPrev" automatically skips back to the body part's previous mode switch keyframe and aligns and keys the controllers of the previous mode to the active mode.
- "Del" deletes all the keys of the body part's active mode on the current frame.

- "DelAll" delete all the keys of the body part's active mode, the mode switch and all the keys of the previous modes on the current frame.

## **Chapter 6: Usability Study**

### **6.1 Methodology**

#### **6.1.1 Survey**

The methodology of the MDP rig usability study was designed for a remote survey of character animator participants of different experience levels. The participants were required to view 5 videos of 3-5 minutes each (one video for each MDP rig part: spine, limbs, IK/DK foot, IK/DK hand and follow/free connectors) and then fill-out a 4-6 question survey for each video. This usability study was cleared by Carleton University Research Ethics Board-B Clearance # 111113 (see Appendix A.1).

The YouTube playlist for the videos can be found at the following URL:  
[https://www.youtube.com/playlist?list=PLEyt0C-1vS0YeFMj0yEjxY\\_D1LvzjvygN](https://www.youtube.com/playlist?list=PLEyt0C-1vS0YeFMj0yEjxY_D1LvzjvygN)

Each video begins with a sped-up demonstration of the manipulation of the respective rig part within the Maya workspace, showing the auto-aligning and keying on each mode switching, then the auto-aligning and key of the previous mode and, finally, the deleting of all the mode switching keyframes. The videos then conclude with a bullet point and verbal explanation of all the controllers and capabilities of the rig part being demonstrated.

The video structure of showing practical usage then explaining the mechanics of each rig part are based on the Constructivism learning theory (teAchnology, 2020) that emphasizes the learning value of "making" over absorbing lessons. Actually, getting character animators to first use the MDP rig themselves was unfeasible – the technical barrier of finding animators with access to a compatible version of Maya and the time to install the MDP plug-in and test the rig was too high. Therefore, the usage demonstration video was

selected as the nearest possible alternative, and as a familiar one to animators, who are used to an online ecosystem of instructional videos as a learning resource.

The survey for each rig part was framed as an "animator opinion survey" and was kept short to encourage participation. The questions concerned interest in the rig part in question, whether the rig part met their animation needs, whether the multiple pivot modes seemed useful, whether the auto-aligning and keying seemed useful and their experience level in animating the body part in question. The "interest" and "need" questions are asked in that order to initially meet the respondents in the subjective "interest" mindset they entered the "animator opinion survey" with, then to shift them to the objective "needs" cataloguing of their personal animation workflow, thus putting them in the appropriate state of mind to consider the usefulness of the multiple pivot modes and the auto-aligning and keying that are then addressed. Each response is in the form of a 5-point Likert scale (Likert, 1932) ranging from negative to positive and little to lots for the experience question. The Likert scale was adopted due to its easily processed numerical values and its widespread use that would ensure a degree of respondent familiarity.

Participants were encouraged to contact the author if they had any questions or concerns about the videos, the terms used, character animation techniques, their own eligibility to participate or the surveys.

### **6.1.2 Participants**

The participants had to be character animators of any experience level that, at the very least, had enough familiarity with a biped character rig that they had animated it in a Vanilla Walk – usually the first biped animation that is taught to animation students (Roberts, 2004). The invitation to participate in the MDP rig usability study was sent to students who

had done animation from Carleton University, the University of Ottawa, Algonquin College and Sheridan College. It was also posted in the "Master's of Rigging" LinkedIn group, the "Animation Mentor", "Anim Props", "Maya Rigging & Animation" and "Animators Assemble!!!" Facebook groups as well as the CG Society animation forum (CG Society, 2020b).

## 6.2 Result Discussion

The survey questions were the same for each rig part:

*Q\_interested*: How **interested** would you be in using this/these [part] rig(s)? (1=Not interested, 5=Very interested)

*Q\_needs*: How much of your [part] animating **needs** does/do this/these rig(s) meet? (1=No needs met, 5=All needs met)

*Q\_pivots*: How **useful** do the multiple pivot modes of the [part] rig(s) look? (1=Useless, 5=Extremely useful)

*Q\_auto-align*: How **useful** does the auto-aligning and keying when switching pivot modes look? (1=Useless, 5=Extremely useful)

*Q\_exp*: How much **experience** do you have animating [part]? (1=Little, 5=Lots)

The limbs had this unique question:

*Q\_IKRoll*: How **intuitive** does the IK Roll controller look? (1=Not at all, 5=Completely)

The IK/DK hand has a unique question concerning the fingers:

*Q\_sticky*: How **useful** do the sticky finger joint modes look? (1=Useless, 5=Extremely useful)

**Table 6.1 Results of the MDP rig usability test.**

Spine	1	5	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Q_interested	Not interested	Very interested	5	5	1	5	3	5	4	2	4	2	5	5	4	4	5	5
Q_needs	No needs met	All needs met	5	5	1	5	4	4	4	4	4	4	5	4	3	4	4	5
Q_pivots	Useless	Extremely useful	5	5	2	5	4	4	5	3	5	4	5	4	4	4	4	5
Q_auto-align	Useless	Extremely useful	5	3	2	5	4	3	5	3	5	3	5	5	4	4	5	5
Q_exp	Little	Lots	3	4	3	5	3	1	2	1	1	2	1	2	3	5	2	4

Limbs	1	5	A	B	C	D	E	F	G	H	I	J	K	L	M	N
Q_interested	Not interested	Very interested	4	1	3	2	4	3	3	4	1	4	5	4	5	5
Q_needs	No needs met	All needs met	5	1	4	2	3	4	4	4	4	4	5	3	4	5
Q_pivots	Useless	Extremely useful	5	1	4	4	4	4	5	4	3	5	3	5	5	5
Q_auto-align	Useless	Extremely useful	4	2	4	3	4	5	5	4	3	5	3	5	5	5
Q_IKRoll	Not at all	Completely	5	1	4	4	4	4	3	4	3	5	4	3	5	5
Q_exp	Little	Lots	4	3	3	2	2	2	2	1	1	1	2	2	4	4

IK/DK Foot	1	5	A	B	C	D	E	F	G	H	I	J	K	L	M	N
Q_interested	Not interested	Very interested	5	1	3	2	5	5	4	1	4	5	3	5	5	5
Q_needs	No needs met	All needs met	5	1	4	2	4	5	4	4	4	5	5	5	4	5
Q_pivots	Useless	Extremely useful	4	1	4	4	5	5	4	2	4	5	4	5	5	5
Q_auto-align	Useless	Extremely useful	4	1	4	3	3	4	4	2	4	5	5	5	3	5
Q_exp	Little	Lots	4	3	3	2	1	2	1	1	1	2	2	5	2	4

IK/DK Hand	1	5	A	B	C	D	E	F	G	H	I	J	K	L	M	N
Q_interested	Not interested	Very interested	5	1	3	2	2	4	4	5	5	4	5	4	5	5
Q_needs	No needs met	All needs met	5	1	4	2	4	5	4	5	5	5	5	4	4	5
Q_pivots	Useless	Extremely useful	5	1	4	3	3	5	4	5	5	5	5	5	5	5
Q_sticky	Useless	Extremely useful	5	1	4	3	3	5	3	5	5	4	5	4	5	5
Q_auto-align	Useless	Extremely useful	4	1	4	3	4	4	3	5	4	4	5	4	5	5
Q_exp	Little	Lots	4	3	3	1	4	2	1	1	2	2	4	4	4	4

Follow/Free	1	5	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q_interested	Not interested	Very interested	5	2	3	2	5	5	4	4	4	5	4	4	5	4	4
Q_needs	No needs met	All needs met	5	2	4	2	4	4	4	4	4	5	4	4	3	4	5
Q_auto-align	Useless	Extremely useful	5	1	4	4	4	4	4	4	3	5	4	5	4	4	5
Q_exp	Little	Lots	4	3	3	1	3	2	1	1	2	2	3	4	1	2	4

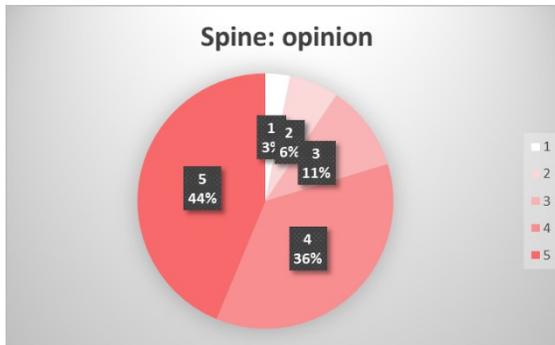


Figure 6.1 Spine opinion level distribution.

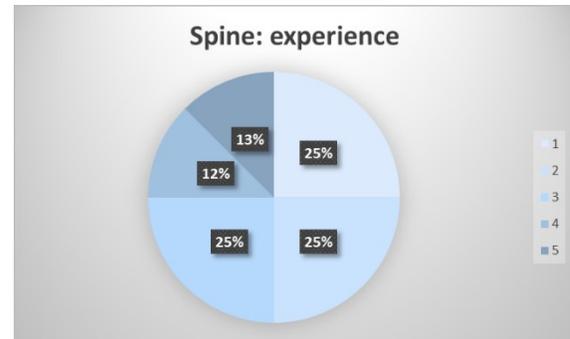


Figure 6.2 Spine experience level distribution.

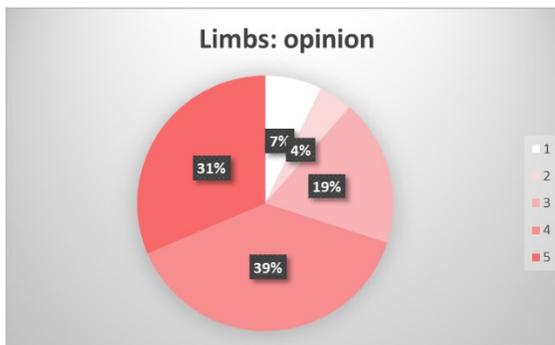


Figure 6.3 Limbs opinion level distribution.

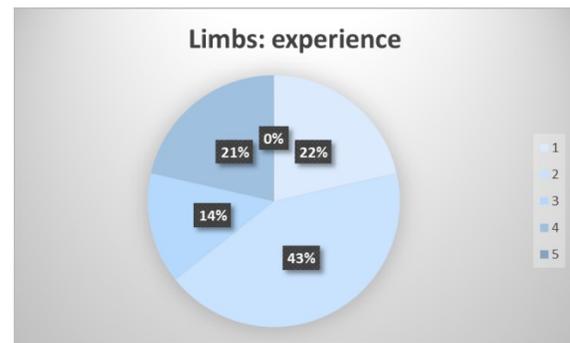


Figure 6.4 Limbs experience level distribution.

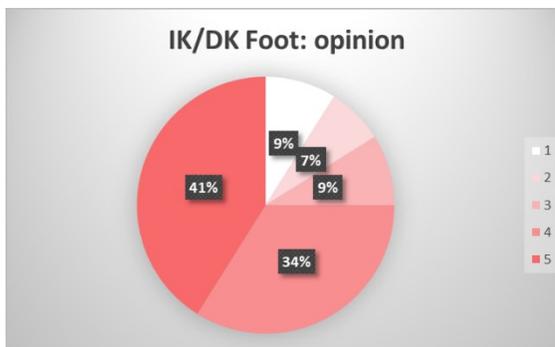


Figure 6.5 Foot opinion level distribution.

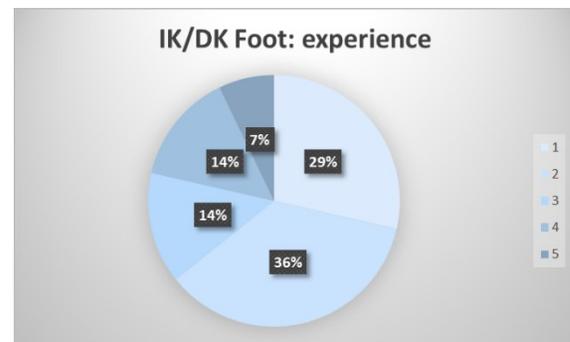


Figure 6.6 Foot experience level distribution.

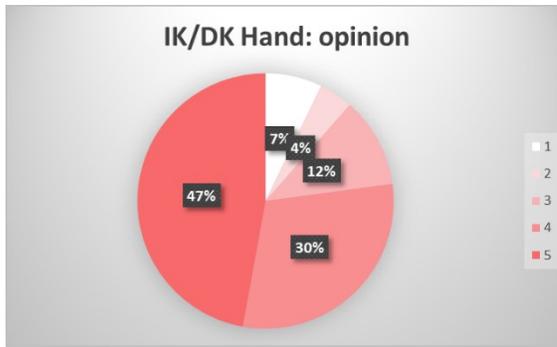


Figure 6.7 Hand opinion level distribution.

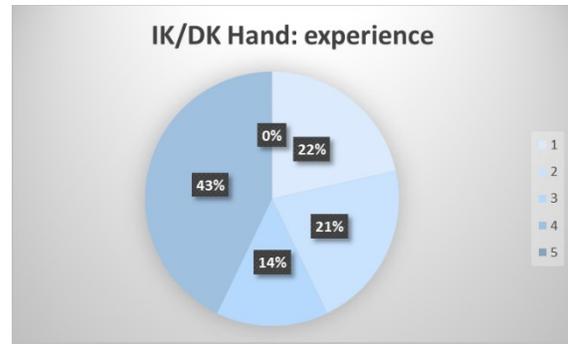


Figure 6.8 Hand experience level distribution.



Figure 6.9 Follow/Free opinion level distribution.



Figure 6.10 Follow/Free experience level distribution.

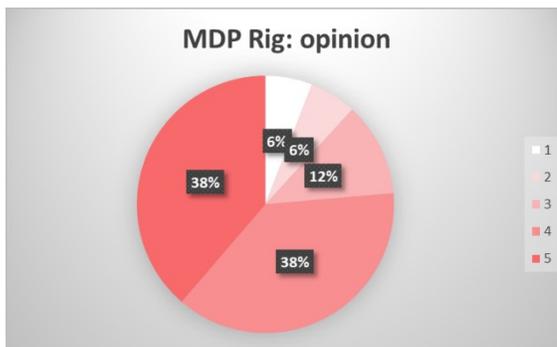


Figure 6.11 Overall MDP rig opinion level distribution.

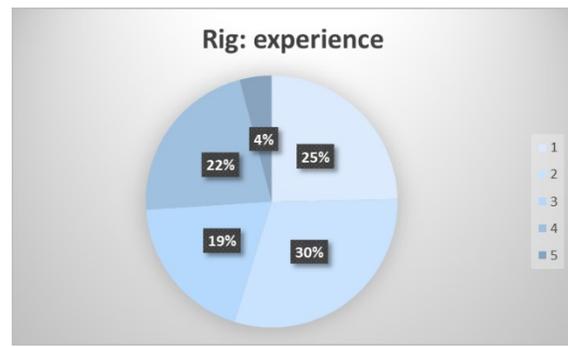


Figure 6.12 Overall rig experience level distribution.

The overall reaction of the character animators to the MDP rig was positive with 38% of reactions in the most positive interested/needs met/useful category and 38% in the second most positive with only 12% being neutral and another 12% being in the negative

interested/needs met/useful categories. The overall experience with character rigs of the participants skews toward the beginner level with 55% self-assessing as being in the less experienced categories, 19% in the middle and 26% greater.

The breakdown of experience by rig part shows the greater experience concentration in the spine and IK feet, which implies a participant pool that has spent much time animating those parts. Perambulatory cycles and jumping tend to be the first animations taught on biped rigs and make much use of the IK feet and spine – the IK legs are also used, but only as an extension of the feet and generally not switched to FK or any other available mode. This would also explain the comparatively *enthusiastically* positive reaction to the spine and IK/DK foot of the MDP rig. Only the IK/DK hand rig proved to be more ardently popular, perhaps due to participants who – having mastered walk cycles – are now animating hands interacting with objects and surfaces and have acquired an appreciation for what the MDP IK/DK hand rig has to offer. This is borne out by the 43% 4th-level experience reported with IK hands but no 5th-level.

The reaction to the MDP limb rig, while by no means negative (more precisely 70% positive with 5th and 4th categories combined) is perhaps explained by their complexity. The sub-rigs of the spine and the IK hands are all variations on the same concept – pivots points along an articulated chain – but the sub-rigs of the MDP limbs are all different. FK and IK are familiar limb sub-rigs and some animators may even have used Double IK but Mid-point IK is new. Furthermore, the MDP limbs also have unfamiliar controllers – the IK Roll controller replaces the familiar Point Vector controller for IK mid-joints – and the upper and lower limb segment controllers are also novel. Add to that the squashing and stretching capabilities of the various controllers and the whole MDP limb rig is the

most conceptually daunting of the MDP rig parts. The 21% 4th-level experience level but no 5th-level supports the lack of deep experience of the participants with animating limbs.

The converse is perhaps the case for the reaction to the follow/free connectors which, again, while not negative – in fact, having greatest overall positive reaction of 82% with 5th and 4th categories combined – has the lowest 5th-level enthusiastic positive level of 27%. This is possibly because the concept of the follow/free connectors are well-known and the MDP auto-aligning and keying between the two modes can also be appreciated. However, it isn't that difficult to blend between the two modes without the MDP auto-alignment, so the MDP follow/free connectors were not deemed as particularly world-changing. The highest mid-level experience of 27% and no 5th-level supports the interpretation that the participants felt that there wasn't much experience to be had with such a simple rig part.

The participants were anonymous, but they could choose to contact the author, so additional feedback from a participant who had a notably negative reaction to the MDP rig (Spine respondent C and Limbs, IK/DK Foot, IK/DK Hand and Follow/Free respondent B) is known. They reported that the controllers were too crowded together which is a concern under Fitts' Law (Fitts, 1954), and that there should be a "Blend" attribute so that the animator could key the blending between the different modes like IK and FK. The participant did not seem to realize that the use of the MDP rig UI automatically aligned and keyed between modes precisely to eliminate the need for animators to manually blend between them. This was an insight into the kind of resistance the acceptance of the MDP

rig may face among animators – especially those who view a custom UI component of the rig – such as the switchUI and selKeyUI of the MDP rig – as optional.

Ideally, the usability test could have had a greater number of participants – at least 50 – and could have included non-animators and 2D animators, as well as 3D, ranging in animation experience from beginner to expert. The testing task would have been to animate a Walk Cycle and a contortion of the body that tested the shifting of the centre of mass, the planting of the limb mid-points and the finger pivots of the hands. These animations would be performed by all testers on both the traditional rig and MDP rig with the testers being recorded and having their gaze tracked on the screen. They would be provided with images describing proposed key poses and on which frame they should occur. They would also be free to ask questions about the rigs. They would then answer a survey about their character animation experience, about their satisfaction using the traditional rig and the MDP rig, and about how intuitive they considered each rig to be to. Having the objective tracking data and subjective survey data would yield a more accurate and nuanced usability result. Non- and 2D animators would either have to be taught the navigational basics of an existing 3D authoring tool, or a specialized character animation tool would have to be created holding only the traditional and MDP rigs and a timeline. The 3D animators would just use the traditional rig they're familiar with first then the MDP rig. Non- and 2D animators' opinions might be coloured by whichever rig they use first, but there might be some insight to be gained from that, so half of non-animators could use the traditional rig first and half the MDP rig. The results of the two non-animator groups could then be compared to ascertain if the first rig they used had an impact on their performance and opinion.

## Chapter 7: Conclusions

The Conclusions chapter summarizes the contributions of the thesis, the limitations and directions for future research.

### 7.1 Discussion of Contributions

The contributions of the thesis are focused on the Multiple Dynamic Pivots rig, which was designed along HCI principles, implemented in Maya and Python and usability tested with a participant pool of 3D character animators. The overall reception of the MDP rig by this sample of intended users was positive. This indicates that the application of HCI to 3D rigging was a success and a worthwhile pursuit for future research.

In conclusion, the *contributions to knowledge* of the thesis are:

- The application of the HCI principles (specifically Direct Manipulation, Affordances and Mapping) to the 3D character rigging and animation.
- The design of the MDP rig adhering to those HCI principles.

The *contributions to practice* of the thesis are:

- The implementation of the MDP rig in Maya using Python.
- The usability testing of the MDP rig by 3D character animators of varying experience levels.

### 7.2 Limitations

The limitations of the MDP rig as designed and implemented in the thesis are as follows:

- The rig was limited to Biped for illustration, but it could be used for other kind of character rigs (such as quadruped and flying creatures).
- The MDP autorig code requires the character model's arms to be in the elbows-bent "hug" pose, which can be more difficult to model than the traditional T-pose.

- The controllers' shapes are insufficiently "grabbable" giving the impression of controller over-crowding around the pivot points.
- The UI button panels take up too much screen space.
- The "Reflect Axis" button in the selKeyUI is incapable of fixing all unintended rotations.
- The anti-candywrapping roll controllers are in a hierarchy that could lead to counter-animation.

### **7.3 Directions for Future Research**

The focus of the thesis was mainly on the design and implementation of the MDP rig in Maya and Python. A future research direction could focus on the implementation of the MDP rig in other 3D tools in order to test design universality. Also, the design could be expanded to include other types of character rigs (e.g., quadrupeds and flying creatures).

Another important direction of future research needs to focus on the design and testing of the controllers and UI for a more intuitive user experience. The controls need to be easily selectable in the 3D workspace, and the UI needs to take-up less screen real estate, while making the relevant functions more readily available to the user. Controller and UI design candidates can be prototyped, and their usability tested to determine the optimal shapes and layouts within the 3D animation space.

## Appendices

### Appendix A Research Ethics Clearance



Office of Research Ethics

503 Robertson Hall | 1125 Colonel By Drive

Ottawa, Ontario K1S 5B6 613-

520-2600 Ext: 4085

[ethics@carleton.ca](mailto:ethics@carleton.ca)

#### **CERTIFICATION OF INSTITUTIONAL ETHICS CLEARANCE**

The Carleton University Research Ethics Board-B (CUREB-B) has granted ethics clearance for the research project described below and research may now proceed.

CUREB-B is constituted and operates in compliance with the *Tri-Council Policy Statement: Ethical Conduct for Research Involving Humans* (TCPS2).

**Ethics Protocol Clearance ID:** Project # 111113

**Research Team: Ms. Mihaela Petriu (Primary Investigator)**

Christopher Joslin (Research Supervisor)

**Project Title:** Bipedal Character Rig Redesign According to Human-Computer

Interaction (HCI)

Principles

**Funding Source** (If applicable):

Effective: **August 26, 2019**  
**31, 2020.**

Expires: **August**

**Please ensure the study clearance number is prominently placed in all recruitment and consent materials: CUREB-B Clearance # 111113.**

**Restrictions:**

This certification is subject to the following conditions: 1. Clearance is granted only for the research and purposes described in the application. 2. Any modification to the approved research must be submitted to CUREB-B via a Change to Protocol Form. All changes must be cleared prior to the continuance of the research.

3. An Annual Status Report for the renewal of ethics clearance must be submitted and cleared by the renewal date listed above. Failure to submit the Annual Status Report will result in the closure of the file. If funding is associated, funds will be frozen.
4. A closure request must be sent to CUREB-B when the research is complete or terminated.
5. During the course of the study, if you encounter an adverse event, material incidental finding, protocol deviation or other unanticipated problem, you must complete and submit a Report of Adverse Events and Unanticipated Problems Form, found here: <https://carleton.ca/researchethics/forms-and-templates/>

Failure to conduct the research in accordance with the principles of the *Tri-Council Policy Statement: Ethical Conduct for Research Involving Humans 2nd edition* and the *Carleton University Policies and Procedures for the Ethical Conduct of Research* may result in the suspension or termination of the research project.

Upon reasonable request, it is the policy of CUREB, for cleared protocols, to release the name of the PI, the title of the project, and the date of clearance and any renewal(s).

Please contact the Research Compliance Coordinators, at [ethics@carleton.ca](mailto:ethics@carleton.ca), if you have any questions.

**CLEARED BY:**  
**2019**

**Date: August 26,**



Natasha Artemeva, PhD, Chair, CUREB-B



Janet Mantler, PhD, Vice-Chair, CUREB-B

## References

- (Aberman et al., 2020a) Aberman, Kfir; Li, Peizhuo; Lischinski, Dani; Sorkine-Hornung, Olga; Cohen-Or, Daniel; Chen, Baoquan (2020), "Skeleton-Aware Networks for Deep Motion Retargeting", SIGGraph'2020, *ACM Transactions on Graphics*, Vol. 39:5, June 2020.
- (Aberman et al., 2020b) Aberman, Kfir; Weng, Yija; Lischinski, Dani; Cohen-Or, Daniel; Chen, Baoquan (2020), "Unpaired Motion Style Transfer From Video to Animation", SIGGraph'2020, *ACM Transactions on Graphics*, Vol. 39:5, June 2020.
- (Autodesk, 2018) Autodesk (2018). "character studio", Retrieved from <http://knowledge.autodesk.com/support/3ds-max/learn-explore/caas/CloudHelp/cloudhelp/2019/ENU/3DSMax-Character-Animation/files/GUID-BC2F93D7-D769-4946-A2B6-3657CBB427A1-htm.html>. Last visited July 2020.
- (Autodesk, 2019) Autodesk (2019). "HumanIK", Retrieved from <http://knowledge.autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2019/ENU/Maya-CharacterAnimation/files/GUID-EDBDA3DB-4715-40EF-9ADF-412F78BFF98E-htm.html>, Last visited July 2020.
- (Bhati et al., 2015) Bhati, Zeeshan; Waqas, Ahmad; Karbasi, Mostafa; and Mahesar, Abdul Waheed. (2015) "A Wire Parameter and

- reaction Manager based Biped Character setup and Rigging Technique in 3DS MAX for Animation”, *International Journal of Computer Graphics & Animation (IJCGA)* Vol.5, No.2, 21-36.
- (Blumenthal, 2016) Blumenthal-Barby, Jennifer S. (2016) Biases and Heuristics in Decision Making and Their Impact on Autonomy, *The American Journal of Bioethics*, 16:5, 5-15.
- (Bowman et al., 2001) Bowman, Doug A; Kruijff, Ernst; LaViola, Joseph J; and Poupyrev, Ivan. (2001) "An introduction to 3-d user interface design," *Presence: Teleoperators and Virtual Environments* 2001, Vol. 10 No. 1, 2001. 96-108, MIT Press.
- (Bowman et al., 2005) Bowman, Doug A; Kruijff, Ernst; LaViola, Joseph J; and Poupyrev, Ivan. (2005). *3D User Interfaces: Theory and Practice*. Boston: Addison–Wesley.
- (Carey, 2020) Carey, Josh; Clark, Brad and Moore, Chad. *Rigging Dojo: Teaching the Art and Science of Character Rigging*, <http://www.riggingdojo.com/>. Last visited July 2020.
- (CG Society, 2020a) CG Society, Workshops, <http://cgsociety.org/workshops>. Last visited July 2020.

- (CG Society, 2020b) CG Society, Animation forums,  
<http://forums.cgsociety.org/c/techniques/animation>. Last  
visited August 2020.
- (Eom et al., 2020) Eom, Haegwang; Han, Daseong; Shin, Joseph S.; Noh,  
Junyong; (2020), "Model Predictive Control With a  
Visuomotor System for Physics-based Character  
Animation", SIGGraph'2020, *ACM Transactions on  
Graphics*, Vol. 39:5, June 2020.
- (Fitts, 1954) Fitts, Paul M. (1954). "The information capacity of the  
human motor system in controlling the amplitude of  
movement". *Journal of Experimental Psychology*. **47** (6):  
381–391.
- (Frohlich, 1993) Frohlich, David M., (1993) "The history and future of  
direct manipulation," *Behaviour & Information  
Technology* 12, 6, 1993, 315-329.
- (Harvey et al., 2020) Harvey, Felix; Yurick, Mike; Pal, Christopher;  
Nowrouzezahrai, Derek (2020), "Robust Motion In-  
betweening", SIGGraph'2020, *ACM Transactions on  
Graphics*, Vol. 39:5, June 2020.
- (Hick, 1952) Hick, W., "On the rate of gain of information", *J.  
Experimental Psychology*, 4, (1952), 11-36
- (Holden et al., 2020) Holden, Daniel; Kanoun, Oussama; Perepichka, Maksym;  
Popa, Tiberiu (2020), "Learned Motion Matching",

- SIGGraph'2020, *ACM Transactions on Graphics*, Vol. 39:5, June 2020.
- (Hutchins et al., 1986) Hutchins, Edwin L.; Hollan, James D.; and Norman, Don A., (1986), "Direct manipulation interfaces;" in Norman, DA. and Draper, SW. (Editors), *User Centered System Design: New Perspectives on Human Computer Interaction*, 1986, 87-124.
- (Hyman, 1953) Hyman, R., "Stimulus information as a determinant of reaction time", *J. Experimental Psych.*, 45, (1953), 188-196
- (Kwon et al., 2020) Kwon, Taesoo; Lee, Yoonsang; van de Panne, Michiel; (2020), "Fast and Flexible Multilegged Locomotion Using Learned Centroidal Dynamics", *SIGGraph'2020, ACM Transactions on Graphics*, Vol. 39:5, June 2020.
- (Lassiter, 1987) Lassiter, John (1987). "Principles of Traditional Animation Applied to 3D Computer Animation". *SIGGRAPH'87. Computer Graphics*, 21(4), 35-44.
- (Likert, 1932) Likert, Rensis (1932). "A Technique for the Measurement of Attitudes". *Archives of Psychology*. 140: 1–55.
- (Lin et al., 2015) Lin, Andy; Lee, Gene S.; Longson, Joe; Steele, Jay; Goldberg, Evan; Stefanovic, Rastko (2015). "Achieving Real-Time Playback with Production Rigs", *ACM SIGGRAPH 2015 Talk*, Los Angeles, CA, July 2015.

- (Ling et al., 2020) Ling, Hung Yu; Zinno, Fabio; Cheng, George; van de Panne, Michiel (2020), "Character Controllers Using Motion VAEs", SIGGraph'2020, *ACM Transactions on Graphics*, Vol. 39:5, June 2020.
- (Luo et al., 2020) Luo, Ying-Sheng; Soeseno, Jonathan Hans; Chen, Trista Pei-Chun; and Chen, Wei-Chao; (2020), "CARL: Controllable Agent With Reinforcement Learning for Quadruped Locomotion", SIGGraph'2020, *ACM Transactions on Graphics*, Vol. 39:5, June 2020.
- (MacKenzie, 1992) MacKenzie, I. Scott (1992). "Fitts' law as a research and design tool in human-computer interaction". *Human-Computer Interaction*. 7: 91-139.
- (Mao et al., 2007) Mao Chen; Qin Shen F; and Wright, David (2007) "Sketching-Out Virtual Humans: A Smart Interface for Human Modelling and Animation". In: Butz A., Fisher B., Krüger A., Olivier P., Owada S. (eds) Smart Graphics. LNCS vol 4569. Springer.
- (Merel et al., 2020) Merel, Josh; Tunyasuvunakool, Saran; Ahuja, Arun; Tassa, Yuval; Hasenclever, Leonard; Pham, Vu; Erez, Tom; Wayne, Greg; Hess, Nicolas; (2020), "Catch & Carry: Reusable Neural Controllers for Vision-guided Whole-body Tasks", SIGGraph'2020, *ACM Transactions on Graphics*, Vol. 39:5, June 2020.

- (Mori, 2012) Mori, Masahiro, "The Uncanny Valley: The Original Essay by Masahiro Mori", Karl F. MacDorman and Norri Kageki (translators), *IEEE Robotics and Automation Magazine*, June 2012.
- (Nielsen, 1994a) Nielsen, Jakob (1994a). "Enhancing the explanatory power of usability heuristics." Proc. ACM CHI'94 Conf. , Boston, MA, April 24-28, 1994, 152-158.
- (Nielsen, 1994b) Nielsen, Jakob (1994b). "10 Usability Heuristics for User Interface Design", Retrieved from NN/gNielsen Norman Group, <http://nngroup.com/articles/ten-usability-heuristics>. Last visited July 2020.
- (NLC, 2001) New Line Cinema, *The Lord of the Rings* (trilogy), 2001-2003.
- (Norman, Don, 2013) Norman, Don. (2013); "The Design of Everyday Things", Revised Edition; Basic Books.
- (O'Hailey, 2013) O'Hailey, Tina (2013). *Rig it Right!: Maya Animation Rigging Concepts*. Publisher Taylor & Francis.
- (Pixar, 1995) Pixar, *Toy Story* (film), 1995.
- (Popa, 1998) Popa, Adrian (1998). "What is meant by the term gimbal lock?", Retrieved from MadSci Network <http://www.madsci.org/posts/archives/aug98/896993617.Eg.r.html>, Last visited July 2020.

- (Preston, 1994) Blair, Preston (1994). *Cartoon Animation*. Walter Foster Publishing, Inc.
- (Ritchie et al., 2005) Ritchie, Kiaran; Callery, Jake; and Biri, Karim (2005) *The Art of Rigging (A Definitive Guide To Character Technical Direction With Alias Maya, Volume 1)*, Publisher CG Toolkit.
- (Roberts, 2004) Roberts, Steve (2004), *Character Animation in 3D*, Focal Press, An imprint of Elsevier.
- (Schleifer, 2006) Schleifer, Jason (2006). "Animator Friendly Rigging". Autodesk® Maya® Master Classes - Instructor Notes, SIGGRAPH'06, Retrieved from <http://jasonschleifer.com/afr/>. Last visited July 2020.
- (Shneiderman, 1983) Shneiderman, Ben (1983). "Direct manipulation: a step beyond programming languages," *IEEE Computer* 16, 8 (August 1983), 57-69.
- (Shneiderman, 1988) Shneiderman, Ben (1988). "We can design better user interfaces: A review of human-computer interaction styles", *Ergonomics*, 31:5, 699-710.
- (Shneiderman, 1997) Shneiderman, Ben (1997). "Direct Manipulation for Comprehensible, Predictable and Controllable User Interfaces", In Proceedings of the 2nd international conference on Intelligent User Interfaces IUI '97, Orlando FL (1997), 33-39.

- (Smith et al, 2012) Smith, Greg; McLaughlin, Mark; Lin, Andy; Goldberg, Evan; Hanner, Frank; "dRig: An Artist-Friendly, Object-Oriented Approach to Rig Building", ACM SIGGRAPH 2012 Talks, August 2012, Article No.: 18.
- (Starke et al., 2020) Starke, Sebastian; Zhao, Yiwei; Komura, Taku; and Zaman, Kazi (2020), "Local Motion Phases for Learning Multi-Contact Character Movements", SIGGraph'2020, *ACM Transactions on Graphics*, Vol. 39:5, June 2020.
- (teAchnology, 2020) teAchnology, "Piaget's Theory of Constructivism", Retrieved from <https://www.teach-nology.com/currenttrends/constructivism/piaget/theories.html>, Last visited July 2020.
- (Thomas et al, 1981) Thomas, Frank; and Johnston, Ollie (1981) *The Illusion of Life: Disney Animation*, Disney Editions, New York.
- (Walt Disney, 2019) Walt Disney, Publications (2019) , <https://www.disneyanimation.com/technology/publications>, Last visited July 2020.
- (Walther, 2014) Walther-Franks, Benjamin (2014), *Direct Animation Interfaces: An Interaction Approach to Computer Animation*, PhD Thesis, Mathematics/Computer Science, University Bremen, Germany.
- (Whited et al., 2010) Whited, Brian; Noris, Gioacchino; Simmons, Maryann; Sumner, Robert W., Gross, M.; and Rossignac, JJarek.

- (2010), *BetweenIT: An Interactive Tool for Tight Inbetweening*. *Computer Graphics Forum*, 29: 605-614.
- (Williams, 2002) Williams, Richard (2002). *The Animator's Survival Kit, 2nd edition*. Faber and Faber, Ltd., p48.
- (Ye et al., 2020) Ye, Hui; Zwan, Kin Cgung; Su, Wanchao; Fu Hongbo (2020), "ARAnimator: In-situ Character Animation in Mobile AR With User-defined Motion Gestures", SIGGraph'2020, *ACM Transactions on Graphics*, Vol. 39:5, June 2020.
- (Zhou, 2017) Zhou, Xiaocheng, (2017), "Application Research of Virtual 3D Animation Technology in the Design of Human Computer Interface," Int. Conference on Computer Network, Electronic and Automation (ICCNEA), Xi'an, 2017, pp. 143-148
- (Ziegler at al., 1988) Ziegler, J.E. and Fahrnich, K. P., "Direct manipulation," In Helander, M. (Editor), *Handbook of Human Computer Interaction*, Elsevier Science Publishers, Amsterdam, (1988), 123–133.