A MULTI-DOF SOFT ROBOT MECHANISM FOR PATIENT MOTION CORRECTION AND BEAM ORIENTATION SELECTION IN CANCER RADIATION THERAPY

by

Olalekan Patrick Ogunmolu

APPROVED BY SUPERVISORY COMMITTEE:

Nicholas R. Gans, Co-Chair

Mark Spong, Co-Chair

Tyler Summers

Dinesh Bhatia

Yonas Tadesse

A MULTI-DOF SOFT ROBOT MECHANISM FOR PATIENT MOTION CORRECTION AND BEAM ORIENTATION SELECTION IN CANCER RADIATION THERAPY

by

OLALEKAN PATRICK OGUNMOLU, BSc, MSc(Eng.)

DISSERTATION

Presented to the Faculty of

The University of Texas at Dallas

in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY IN

ELECTRICAL ENGINEERING

THE UNIVERSITY OF TEXAS AT DALLAS

August 2019

ACKNOWLEDGMENTS

Special thanks my advisor, Nick Gans, for his mentorship throughout my graduate degree. He encouraged me to explore my ideas and gave me the freedom to carve my own research niche as time wore on. I am thankful for his infinite patience, kind corrections, and technical prowess. I am indebted to Professors Mark Spong, Phillip Anderson, Yonas Tadesse, Tyler Summers and Dinesh Bhatia, members of my dissertation committee, for their invaluable feedback. Dr. Tadesse, alongside his students, were helpful in the design of the 3 *degrees-of-freedom* head and neck testbed described in this work. The time I spent in Dr. Summers' lab in my third year exposed me to game theory and its applications to designing robust approximate dynamic programming policies.

My work would be less effective without the support of my friends and family. I am particularly grateful to Audrey Sedal for her insightful feedback on the continuum models presented in Chapters 2 and 3. Ireti Akinola had the patience to correct errors in my derivations. Biodun, my little brother, was thoughtful enough in his critique. I am indebted to these folks and many others for their invaluable feedback. Thanks to Ebby Oladimeji for his mentorship and friendship. I conducted the beam orientation optimization part of this work at UT Southwestern Medical Center's MAIA Lab. The many interactions I had with fellow graduate students, postdocs, and professors widened my intellectual horizon. I am thankful (in alphabetical order) to Anjali Balagopal, Michael Folkerts, Yesenia Gonzalez, Xuejun Gu, Weiguo Lu, Xun Jia, Ana Montero, Dan Nguyen, Azar Sadeghnejad, and Chenyang Shen, and most importantly my UT Southwestern advisor, Steve Jiang inter alia, who made coming to the lab something to look forward to daily.

I was fortunate enough to work with delightful office mates in RoboTec. Thanks (in alphabetical order) to David Allen, Terrell Bennett, Edgar Bolivar, Kyle Embry, Kaveh Fathian, Jon Horn, Bashir Jafari, Asif Iqbal, Namhyung Lee, David Quintero, Pablo Ramirez, Venkatraman Renganathan, Sleiman Safaoui, Yujie Zhang and many others for their academic camaraderie.

May 2019

A MULTI-DOF SOFT ROBOT MECHANISM FOR PATIENT MOTION CORRECTION AND BEAM ORIENTATION SELECTION IN CANCER RADIATION THERAPY

Olalekan Patrick Ogunmolu, PhD
The University of Texas at Dallas, 2019

Supervising Professors: Nicholas R. Gans, Co-Chair
Mark Spong, Co-Chair

Accurate patient immobilization in conformal radiation therapy is crucial for efficient cancer treatment. Good treatment outcomes require accurate *patient immobilization* and a good choice of *beam orientations*. State-of-the-art immobilization systems rely on metallic or rigid masks which lack morphological properties, attenuate ionizing radiation, degrade dose efficacy, and are uncomfortable for the patient during treatment. The de-facto *open-loop* and *deferred* positioning procedures sometimes cause eczema or brain damage. We synthesize system identification, finite elastic deformation, and control systems to harness soft robot mechanisms for *real-time* motion correction in cancer radiation therapy scenarios.

Additionally, in most *inverse treatment planning* schemes today, the "right" beam angles among the myriad possibilities in beam space are usually determined through intuition and experience by treatment planners in a time-consuming trial-and-error procedure. Existing mathematical optimization techniques fail to meet a *(near) real-time* planning requirement. We propose a supervised pre-training of a deep neural network to assure quality beam plans are predicted in a real-time feasible manner. Our approach has the advantage of predicting feasible beam angles in near real time, and it is adaptable to treatment modalities that require large beam plans, and 4π-noncoplanar radiation therapy such as VMAT.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

Cancer is a set of related diseases that continually splits some of the body's cells and invades surrounding healthy tissues in a process that mitigates the healthy growth and division of new cells. Cells in a healthy body grow and divide as the body needs them. In a healthy body, as cells become old, they die off and new cells are formed. When cancer is present in the body, however, old cells stick around, becoming damaged over time, and newer cells are unnecessarily produced. As the tumors spread, mitosis occurs, and the new cells travel to other parts of the body via the blood or lymphatic nodes and form new tumors – replicating themselves in a distributed fashion. Distributed cancer is very difficult to treat with traditional methods such as chemotherapy or surgery: chemotherapy destroys benign cells along with malignant cells, and surgery is too painstaking for the various locations where the tumor might have formed in the patient's body.

Means of treating cancers include drugs, immunotherapy, targeted therapy, stem cell transplant, precision therapy, radiation therapy, chemotherapy, and surgery. In this dissertation, our focus is on the use of radiation therapy (RT) in cancer treatment. RT is an active area of research with great potential for improving therapy and reducing the toxicity of radiation. It encompasses the use of ionizing radiation as electron beams, photon beams or x-rays, conformed to a high-energy dose-volume and shaped into the geometry of a tumor so as to render malignant DNAs necrotic while sparing healthy cells. The radiation is produced by a moving linear accelerator machine (LINAC), while precision-targeting and cross-firing of radiation beams from multiple directions creates an ablative surgical procedure by which radiation beam destroys tumors.

A computer-controlled *multi-leaf collimator* (MLC), shown in Figure 1.1, constructed from tungsten due to its high absorption properties for radiation, focuses and shapes the geometry of the radiation beam in order to accurately target tumors. During the treatment planning procedure, a planner may have to delineate the organs-at-risks (OARs) and three dimensional (3D) target volumes; the 3D volumes are concatenations of 2D slices of the patients' CT image. This is so that

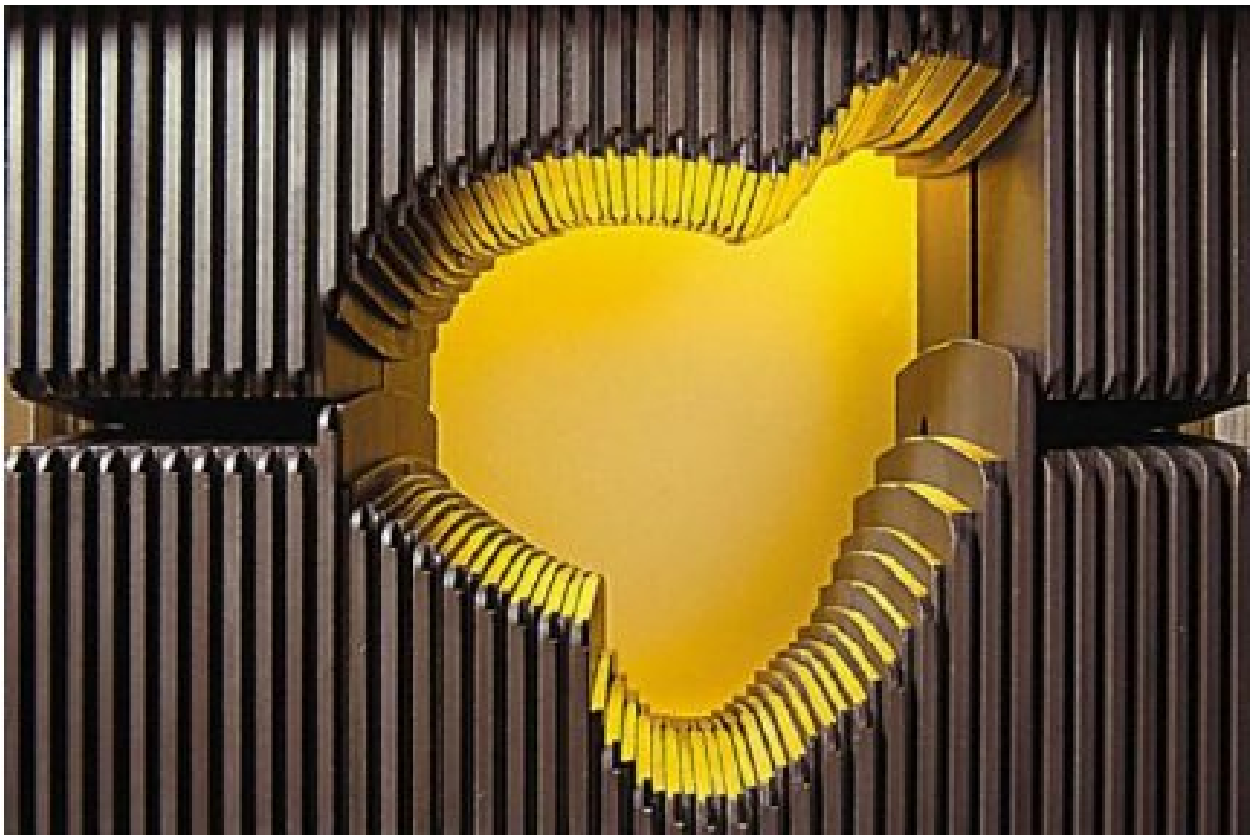


Figure 1.1: A multi-leaf collimator (MLC) used in IMRT and 3DCRT. ©Varian Medical Systems.

a physician can unambiguously define treatment objectives. Before treatment, a patient needs to be properly calibrated – to sub-millimeter translational and sub-degree geometrical accuracy – on a treatment couch (see Figure 1.7). Studies have shown that the rotational angle misalignment between a patient's pose and a registered dose that conforms to the tumor of interest often affects dose delivery efficacy, which in turn reduces the quality of a treatment (Takakura et al., 2009; Xing, 2000a).

IMRT is a cancer treatment method that delivers geometrically-shaped, high-precision x-rays or electron beams to tumors by modulating the intensity of the radiation beam. A multileaf collimator shapes a conventional geometrical field, and the intensity of the geometric field shape is varied bixel-wise in order to modulate the "fluence" (influence matrix) profile around a tumor. This is done while the patient lies in a supine position on a treatment table. Before an IMRT treatment is scheduled, *critical structures* (or tumors) within a *target volume*, and OARs are contoured. Doses

that must be delivered are then prescribed . Each beam to be delivered consists of beamlets, aimed from the same angle, where each beamlet may be of a different intensity from that of its neighbors. Radiation intensities may be delivered from about $5 - 15$ different beam orientations with multiple collimator units. The process of choosing what beam angle is best for delivering beamlet intensities is termed *beam orientation optimization* (BOO) while the process of determining what intensity meets a prescribed fluence profile by a doctor is termed *fluence map optimization* (FMO).

1.1 Treatment Planning and Radiation Therapy

The treatment planning process in radiation therapy involves the careful and deliberate employment of available tools and treatment procedures needed to realize a desirable cancer treatment outcome. The radiation therapy procedure in itself does not necessarily guarantee a favorable outcome without the skillful and careful exertion of all available treatment options necessary for harnessing the potential benefits of radiation therapy (Khan et al., 2016). Treatment planning encompasses finding the optimal *parameters* of a treatment. These parameters may include *dose-limiting structures*, OARs within a *target volume*, doctor's *dose prescription*, *dose fractionation*, *dose distribution*, *patient positioning* on the treatment machine, and the machine's parameters. These parameters are carefully specified as a treatment blueprint, which is to be precisely followed over several weeks of frationated treatment procedure.

1.2 Treatment Planning Parameters

In this section, we will examine the parameters of the treatment planning process that will allow us to answer the research questions posed in this work. What follows is not an exhaustive overview of the parameters related to treatment planning but only those ones that concern this work. Readers are referred to (Khan et al., 2016) for a detailed exposition on treatment planning.

Target volume. As soon as radiation therapy is determined to be the treatment modality for a particular patient, the location and spread of the tumor need to be thoroughly delineated. OARs

and *critical structures* within a *target volume* are contoured slice-wise in the 3D volume so that a doctor can *prescribe* doses that must be delivered to each structure prior to treatment. A crucial aspect of the treatment planning process is the local or regional control of a tumor. Localization of a tumor may be achieved via computed tomography (CT) or magnetic resonance imaging (MRI) scan, ultrasound, single photon emission computed tomography (SPECT) or positron emission tomography (PET) (Khan et al., 2016; Webb, 2001). The outline of the microscopic and visible portion of the tumorous cells in a patient is the *target-volume*. This contains the tumor and its possible spread to regions surrounding the tissues. The most sophisticated imaging devices do not often reveal the geometric details of the microscopic spread of a tumor to surrounding tissues. The part of the tumor that is visible under the imaging device is the *gross target volume* (GTV), while the invisible microscopic part of the tumor, typically estimated during treatment planning, is referred to as the *clinical target volume* (CTV). If microscopic diseases (invisible to imaging devices) are missed during contouring, one may encounter radiation therapy failure, which is undesirable. Since CTV boundaries are estimated, their spatio-temporal location are constantly changing, dependent on respiratory motion, internal displacement of body organs, or treatment setup. Therefore, a geometrical *planning target volume* (PTV) is often defined to contain the CTV within a margin of error in order to account for treatment planning under uncertainty. The PTV is the recommended target volume by the International Commission on Radiation Units and Measurements (ICRU) for guaranteeing a clinically acceptable probability that an adequate dose will be delivered to all parts of the CTV. ICRU defines the CTV and GTV as purely oncological concepts, while the PTV accounts for patient motion and inaccuracies in patient beam positioning. Figure 1.2 illustrates these concepts as defined by the ICRU.

Isodose distribution. In practice, the isodose distribution is determined by iteratively optimizing doses so that tumor cells are (ideally) completely irradiated with the right amount of dose whilst sparing normal tissues. The complexities of balancing the physical constraints of dose delivery machines, the uncertainty of the CTV coverage region and the specific characteristics of beams may

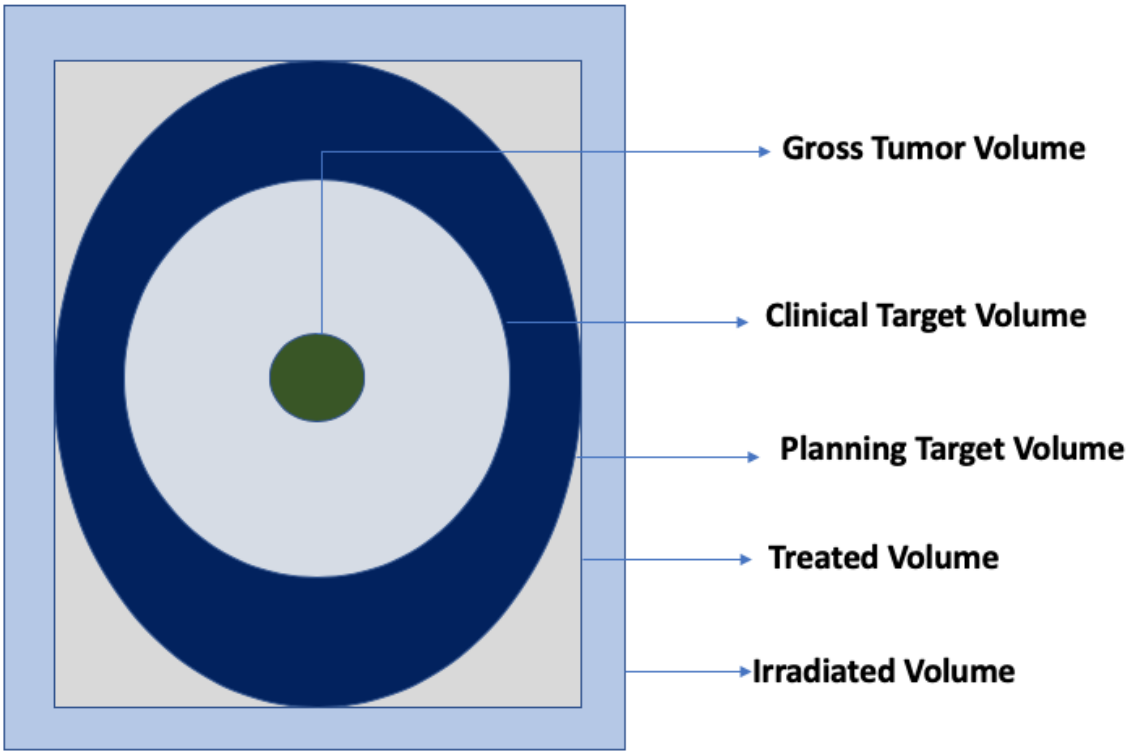


Figure 1.2: Schematic of the various components that constitute an ICRU volume.

mean that certain healthy cells be sacrificed at the expense of providing an ideal dose distribution. Isodose planning includes the optimization of beam weights, beam modifiers, beam placement and beam energy. IMRT optimizes isodose distribution in a reverse engineering process where a uniform conformal target dose spares OARs from incident beam intensities in a technique called *inverse planning*.

Intensity Modulation. Conventional radiotherapy seeks to destroy malignant cancer cells by targeting a uniform-intensity profile of high energy x-rays on cancer tumors while minimizing the amount of radiation to normal cells. Uniform intensity is achieved with rectangular fields, blocks, and wedges to specify flatness and symmetry (Webb, 2001). While useful, conventional radiotherapy may cause unwanted effects on surrounding tissues. Three-dimensional conformal radiation therapy (3-DCRT) leverages the advantage of multi-leaf collimators (MLCs) by engineering geometric field-shaping to avoid sustained damage to normal tissues during radiotherapy. It matches the spatial

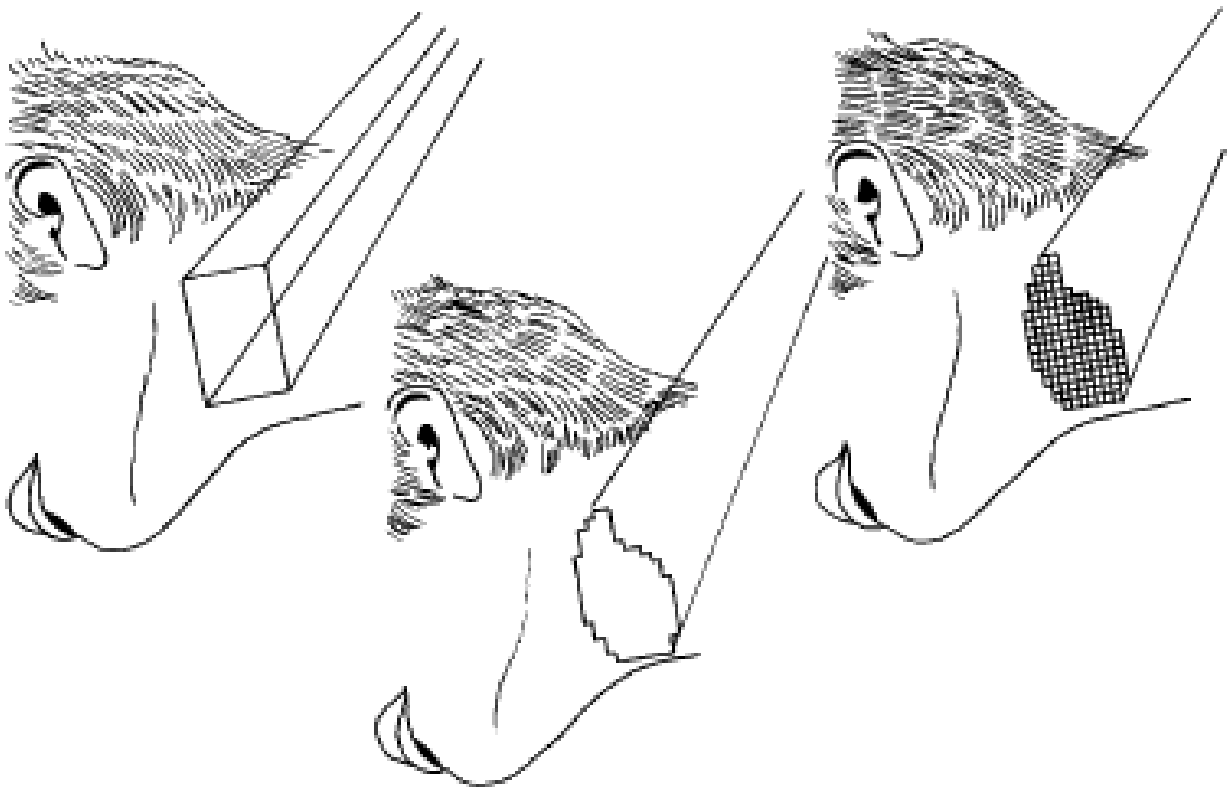


Figure 1.3: [Left - Right]: Conventional radiotherapy, conformal radiotherapy (CFRT) without intensity modulation, and 3D-CRT with intensity modulation. Reprinted from (Webb, 2001).

localization of a high-dose volume to the target volume while avoiding normal tissues. A further advancement in 3-DCRT is *intensity-modulated radiation therapy* (IMRT). IMRT constitutes a broad class of cancer treatment modalities where local control of tumor is improved by controlling an external beam's physical delivery. Advancements in CT technology enables the shaping of internally uniform fields with MLCs that conform the projection of a treatment target volume onto a radiation beam that is then made incident on a tumor (Boyer et al., 1992). Essentially, a MLC shapes a conventional geometrical field and varies its intensity bixel-wise so that the fluence (influence matrix of the incident radiation) intensity is modulated. This has proven useful, especially when the target volume has a concave surface or is close in distance to organs at risks (OARs) (Webb, 2001). Figure 1.3 illustrates the geometrical properties of these three forms of radiation therapy treatment.

1.3 Robotic Radiotherapy and Patient Positioning Mechanisms

Robotic radiation therapy is a form of radiation therapy that seeks to accurately irradiate a moving patient and a moving target with the aid of robots (Schweikard et al., 1995; Webb, 1999, 2001). This may involve using multiple degrees of freedom (DOF) robotic arms to overcome the limitations of cylindrical radiation beams of fixed cross section in delivering radiation (Schweikard et al., 1995). A rotating linear accelerator gantry is mounted on a robot's end effector, thereby generating high-energy photons in order to create high-dose volumes of radiation (Webb, 2000). A patient positioned on a 6-DOF translational and rotational robotic couch for motion alignment gets the incident radiation at appropriate times during treatment (Gevaert et al., 2012; Lee et al., 2012). By developing a geometric method for planning the beam's adaptable shape as well as the beam motion, treatment time can be evaluated as well as the transparency of the interactive treatment planning. The robot aims radiation beams given any orientation relative to the target volume, thus giving IMRT greater flexibility than most traditional radiation delivery methods. Alongside IMRT, image guided radiotherapy (IGRT) is used to assure the precision of dose targets. IGRT methods employ ultrasound, 3D imaging systems, 2D X-ray devices and/or computed tomography to instantly amend positioning errors, and improve daily radiotherapy fractions' precision. Image guidance in the radiation therapy process is an important tool in the treatment value chain that enables advancements in the delivery of improved dose distributions (Verellen et al., 2008).

In order to avoid dose miss, guarantee precision of dose delivery, repeatable positioning when escalation of dose is necessary in a target volume or when OARs' exposure to toxicity need be minimized, a patient's position on the treatment machine should not fluctuate. In a geometric miss, for instance, highly conformal potent dose increases the risk of underdose to tumors or undesirable high dose to critical organs and nearby tissues. Studies have shown that patient displacement and beam angle misalignment during IMRT produced a 38% decrease in minimum target dose or 41% increase in the maximum spinal cord dose (Xing, 2000b). While image-guided radiotherapy (IGRT) has improved IMRT accuracy while reducing set-up times (Ahn et al., 2009; Robb et al., 2013;

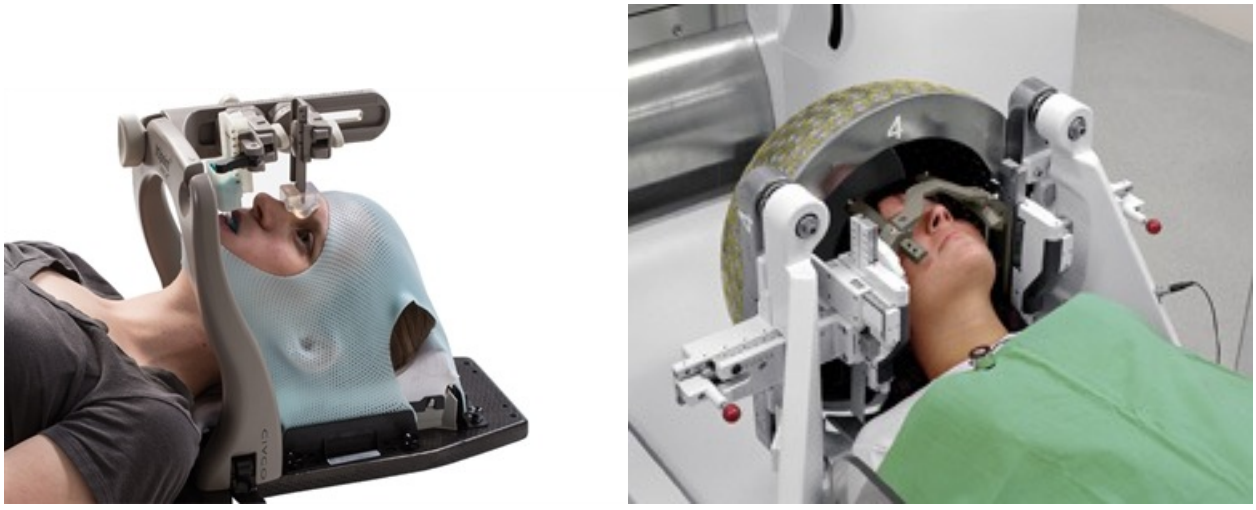


Figure 1.4: Masks and frames used for head immobilization in IMRT

Takakura et al., 2010), current IGRT practices focus on using images acquired before treatment to confirm beam placement (Jaffray, 2012). As such, rigid frames and masks (see Figure 1.4) are used to keep the patient immobilized on the machine so as to mitigate these errors. However, the discomfort caused by head and neck masks and frames in prolonged IMRT treatment can increase patients voluntary and involuntary motion. Studies show that translational errors caused by patient motion can be larger than 6mm, and rotational errors can be as high as 2° (Kang et al., 2011). These can compound over the treatment duration with potential harmful effects. More so, these rigid positioning systems are time-consuming to calibrate on a treatment machine since doses are usually delivered in fractions over many weeks or months: each time a patient comes to the clinic, they need to be re-calibrated on the machine, as it is impractical for patients to wear frames/masks in and out of clinics during elongated periods of treatment. Rigid frames and masks lack real-time position correction of patient's head motion and have been known to cause patient discomfort after treatment, in spite of their strong stiffness properties. When involuntary motions occur during treatment, post-treatment disorders may arise such as eczema, brain lesions and tissue toxicity (Takakura et al., 2009; Xing, 2000a).

Setup errors (interfractional) or patient motion (intrafractional) errors often need to be accounted for during RT. While intrafractional errors can be minimized by highlighting the importance of

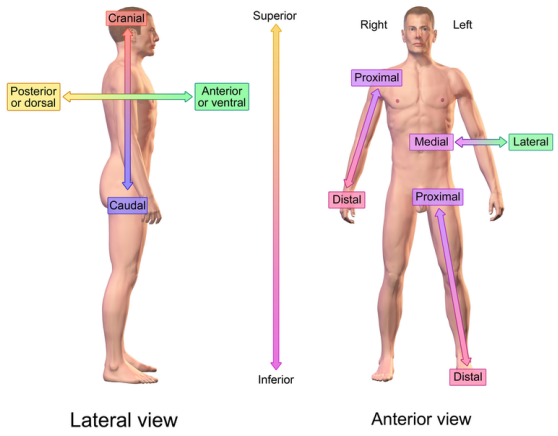


Figure 1.5: SI anatomical axis location. Reprinted from Wikipedia.

voluntary stillness to the patient, suitable means of immobilization and adaptive positioning are necessary when the patient moves involuntarily or sleeps. A good immobilization system should assure precise and accurate targeting of critical organs, ensure dose delivery precision whilst keeping the patient comfortable during treatment. Frameless and Maskless (F&M) RT is promising because it minimizes invasiveness and reduces setup times while comfortably positioning the patient. Before we review the state-of-the-art in frameless and maskless radiation therapy, we briefly introduce the correspondence between the axes of motion used in anatomy and how they correspond to rotational motion in robotics.

1.3.1 Anatomical Motion Axes of the Human Body

In anatomical taxonomy relating to radiation oncology, three major axes are generally employed to describe the relative locations of organs in the human body, namely the *left-right (LR)* axis, the *anterior-posterior (AP)* axis, and the *superior-inferior (SI)* axis. A complete motion about these three axes define a complete 6-DOF motion of a part of the human body. The *posterior-anterior* axis runs from the front of the body through the back of the body, while the *superior-inferior* axis runs from the cranial location of the head through the pelvis region of the body. These are both illustrated in Figure 1.5.

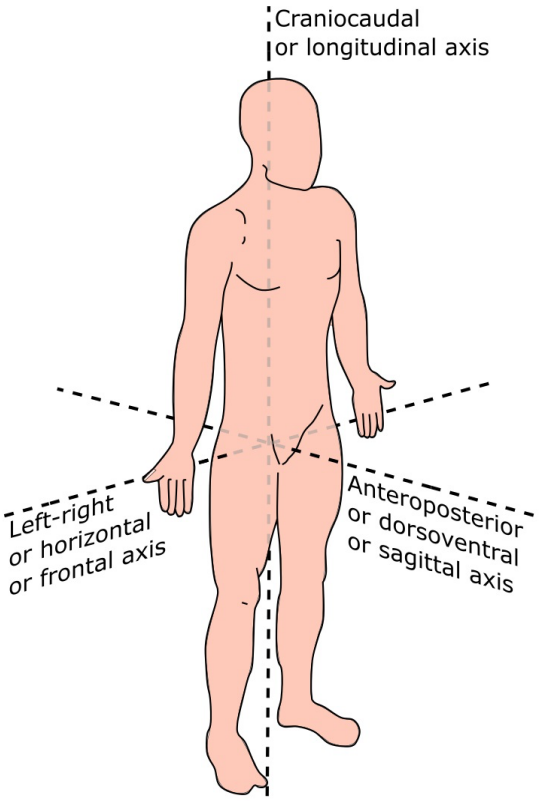


Figure 1.6: LR anatomical axis. Reprinted from Wikipedia.

Rotations around the LR, SI, and AP axes respectively describe motions along the yaw, roll, and pitch axes respectively in the aircraft principal axes coordinate system. These three axes are attached to the body and move as the body executes torque motions along each axial direction. Finally, the left-right axis runs laterally from left to right across the human body, as illustrated in Figure 1.6.

1.3.2 Frameless and Maskless Radiation Therapy

During radiosurgery, it is important to keep the patient immobilized on the treatment machine in order to avoid translational and rotational errors. Accurate control is required so that healthy tissue around a tumor is not excessively irradiated to the point where such tissues are damaged or killed. *Frame-based stereotaxy* immobilizes regions of the patient's body and head so that sensitive organs be properly irradiated to render, e.g. , brain tumor cells necrotic. Accomplishing this typically

involves positioning the patient on the treatment bed while a frame is securely attached to the patient's head. Such treatments are not suitable for fractionated, small doses, which are repeatedly given to the patient from a few weeks to a couple of months. These fractionated doses are necessary when previously irradiated tissues are required to heal while other organs are irradiated. It becomes impractical to leave the frame on the patient's head, since treatment usually lasts several rounds for weeks or months. The complex process of removing and reattaching the frame would result in different positions of the frame each time – defeating the purpose of accurate positioning. Therefore, while frame-based stereotaxy gives desired accuracy, it is not feasible for fractionated treatment plans.

To circumvent attaching frames with screws over the cranial region of the patient as well as the need to focus radiation on regions outside the cranium, *frameless stereotaxy* was developed (Murphy and Cox, 1996). In frameless stereotaxy, a 3D patient mapping is created for the body region of interest and stored away as reference. Two or more diagnostic beams are then passed through the mapping region where the beams are at predetermined non-zero angles from one another. At predetermined time intervals, the stored and diagnostic images are compared to reference data, and the error between them is used to adjust the beam used on the target region; this ensures alignment with the dose of the surgical prescription.

Frameless and maskless positioning systems aim to immobilize the patient without the rigid masks and frames that are often fraught with patient discomfort and usually lack real-time position correction. We briefly review related works that attempt to control a patient's head and neck motion in frameless and maskless scenarios in radiation therapy. We then provide a general overview of soft robot actuators in subsection § 1.3.3, describe the typical hardware designs in § 1.3.4 and modeling/control approaches in § 1.3.5.

Initial research into F&M radiation therapy procedures consisted of investigations of soft immobilizing devices around the patient's head and neck region without a real-time controller in the loop. These techniques were mostly evaluative studies meant to investigate the feasibility of

non-rigid immobilization techniques in treatment planning. We provide a general overview of prior investigative studies on these techniques in what follows.

Cervino et al. (Cerviño et al., 2012) fitted expandable foams around the sides and the back of a patient's head while leaving the face exposed. Essentially, the immobilization mechanism was a customized head mold that minimally immobilized the patient while leaving the face free. Patient set-up was performed using computed tomographic (CT) scans before treatment. Simulated treatment accuracy showed an average treatment time of 26 minutes, with patients who slept during experiments taking longer as a result of involuntary movements. In (Cerviño et al., 2010), the authors evaluated the accuracy of a head mold that minimally immobilized a patient's H&N region while leaving the face free in a controlled positioning experiment with volunteers. A 3D surface reconstruction imaging system was used in monitoring patients' position, and treatment was stopped whenever motion exceeded a defined threshold. While the monitoring system showed great clinical accuracy, it assumed and required high patient cooperation in order to achieve the desired immobilization. (Murphy et al., 2008) analyzed the position and velocity components of a Calypso electromagnetic localization system in order to capture *normal free breathing*. In (Navarro-Martin et al., 2015), the authors assessed the differences in interfractional setup accuracy in stereotactic radiation therapy by evaluating the mean displacements for a vacuum cushioning system and thermoplastic masks. With tests on 73 patients for a total of 246 cone beam CTs, it was found that thermoplastics provided a lower deviation in average displacement compared to the vacuum cushioning system. Using a head mold, an open face-mask, and a mouthpiece, (Li et al., 2015) quantified the residual rotation and positioning errors in an open-loop setting to ascertain the reduction in setup time during patient positioning setup. The head mold and open face mask system restricted head motions to within $0.6^\circ \pm 0.3^\circ$ with the time spent on motion corrections limited to 2.7 ± 1.0 min. With advances in mechanical designs and image-guided adaptive techniques, radiation therapy is increasingly witnessing the incorporation of real-time imaging and positioning devices to manipulate patient motion (Krauss et al., 2011).

Real-time patient-positioning systems employed in F&M radiotherapy until now have been mostly rigid electro-mechanical links connected at discrete joints. Robotic stages are generally used in correcting motions of test patients and manikin phantoms in a controlled setting. Example implementations include steel-cast assembled4-DOF robot link components (Liu et al., 2015), HexaPOD parallel manipulators (Herrmann et al., 2011), or in-house fabricated Stewart-Gough platforms such as proposed by (Belcher, 2017). Approaches in feedback control typically employ a vision-based sensing system in monitoring and measuring real-time patient motion. The vision systems may include one or all of infra-red markers, polaroid sensors or a highly sensitive vision tracking system such as the AlignRT (London, UK) 3D vision sensor[1] to measure deviation of a tumor or the isocenter of the patient from a reference configuration. These measurements are then used in a feedback control loop to control the patient's motion or tumor motion.

The robotic stage of (Liu et al., 2015) consisted of 4-axis motor controllers and a power amplifier. The authors derived a 6-DOF calibration framework that transformed the position of a patient head into the coordinate system of an IR tracking camera. The vision-based setup was then used in a real-time feedforward and feedback control of the translational (x, y, z) and the rotational (pitch) motion of a phantom and some human trials; the optical sensing system tracked the pose of the head while a decoupling control law regulated the xyz-translational and pitch motions of the head. A stepper motor controlled the motion of the pitch axis about a pivot point at the base of the mechanical platform. The authors reported that they achieved a correction accuracy of 0.5mm along the $L - R(X)$, $S - I(Y)$, $A - P(Z)$ axes and a 0.2° rotational accuracy about the pivot to the Z-translation axis. (Herrmann et al., 2011) utilized a 6-DOF robotic HexaPOD treatment couch for motion compensation of lung tumors. Leveraging the fast and precise positioning of heavy payloads, the authors implemented a linear auto-regressive exogenous parameter-identification system to identify the HexaPOD's dynamics; a model predictive controller then manipulated a lung tumor with respect to a treatment beam under low frequencies to accomplish real-time motion

[1] AlignRT: https://www.visionrt.com/product/alignrt.

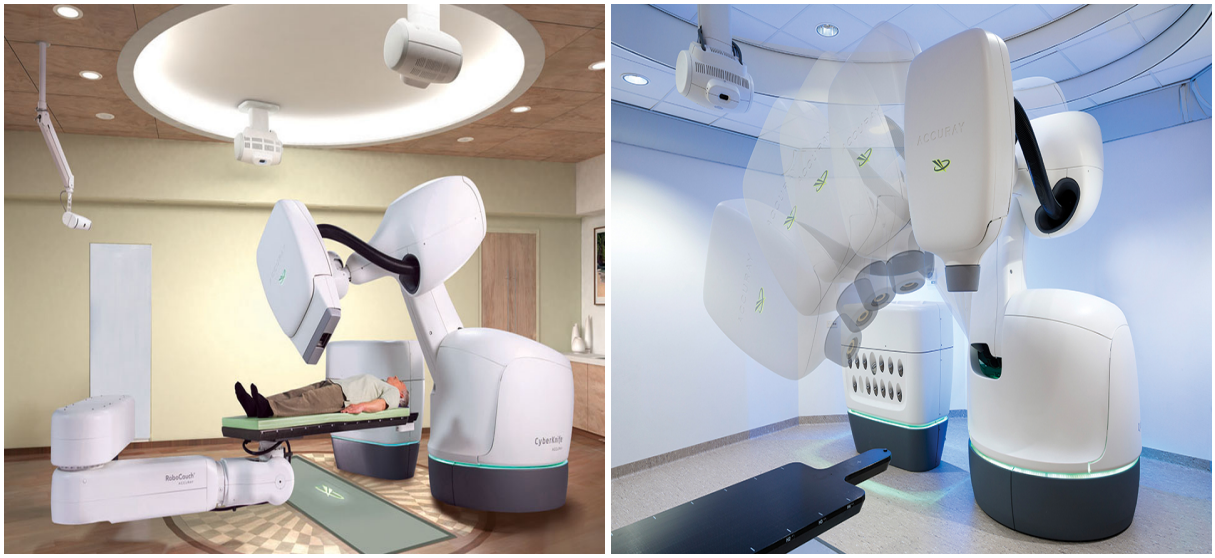


Figure 1.7: The Cyberknife radiation delivery and 6-DOF robotic couch systems. ©Accuray Inc.

compensation. (Haas et al., 2012) used an Elekta 4-DOF (3 translation and one rotational) parallel robot to first simulate and then control couch-based motion in real-time. The authors used a linear state-space model to approximate the rigid body dynamics of the patient support system earlier proposed in (Haas et al., 2005). The state-space model, and an asymmetric dead zone model of the patient support system predicted patient motion, while the velocity response of the patient to voltage input was implemented in a lookup table between a nonlinear gain and a second-order transfer function with asymmetric deadzone. Finally, a Kalman filter acted as a redundant motion prediction for the computationally complex prediction model scheme described. Afterwards, a PID controller was implemented to achieve necessary control along the relevant axes of motion for the system.

There are advances in motion-tracking systems in industrial systems such as Cyberknife and Novalis (see Figures 1.7 and 1.8). Cyberknife ensures complete non-invasive radiotherapy by using implanted tiny gold fiducials to differentiate tumors from healthy tissues. These machines employ a light-weight linear accelerator fitted to a robotic arm in order to direct the beam along 6-DOF . These machine systems are not compatible with the conventional linear particle accelerators used at the majority of cancer treatment centers, they are very expensive to use, and they are limited in consistent treatment planning given their basic assumption that a patient's body is rigid during

Figure 1.8: Noninvasive RT. ©Novalis.

motion tracking and compensation. Being non-compliant, they are incapable of safe interaction with human beings, and it is typical in manufacturing environments, where they are widely used, to separate human workspaces from rigid robotic workspaces. The human body is a natural system that needs to be manipulated with materials that can absorb much of the energy arising out of collisions, rather than rigid electro-mechanical components. We therefore explored the integration of soft-bodied robots made out of compliant materials in building our actuation system for cranial manipulation in robotic radiotherapy.

A major drawback of most of these systems is that they do not address the attenuation of ionizing beams during treatment caused by the rigid electro-mechanical systems that they employ in immobilization. The presence of EM stages can significantly reduce the intensity of incident radiation. One of the goals of this dissertation is to investigate the effectiveness of *radio-transparent* soft robots in IMRT in order to provide real-time motion correction for patients in RT whilst guaranteeing patient comfort.

1.3.3 Soft Elastomeric Actuators: An Overview

The approaches described in this section involve the manipulation of head and neck phantoms on a treatment table using radio-transparent soft actuators that exhibit extensional deformation compression. Continuum-based models are described in the next chapter where we analyze and synthesize the homogeneous deformation of the soft elastomers.

Soft robots or soft actuators are deformable enclosures typically filled with fluid membranes to enable manipulation or locomotion tasks by controlling the fluid in the robot's chamber. They generally exhibit a distributed deformation when they are transformed between configurations. This means they exhibit infinite DOFs , possess hyper-redundancy in their configuration space, thus making them capable of flexible manipulation wherein the tip-point of the actuator can achieve multiple shapes and configurations in a three-dimensional workspace. Given their minimal or non-resistance to compressive shear strain, their compliance make them suitable for biomedical applications where delicate manipulation may be required, as opposed to rigid mechanical platforms that exhibit a high load-to-weight ratio coupled with high stiffness – impractical in enabling articulation of human body parts. Soft robots have the capability of bending, deforming, and twisting with high curvatures, thus enabling their use in confined spaces and radio-transparent required manipulation settings. If carefully designed, they can continuously deform their bodies and emulate biological motions, as well as adapt their geometry to an environment – employing their embodied intelligence and morphological computation property in order to manipulate objects (Zambrano et al., 2014). The shape, geometry, location, and compliance properties of an organism's body parts that exist in nature define the perception and environmental interaction that enable the synergistic *morphological* connection of these features with expressed behaviors (Zambrano et al., 2014). This morphological computation enables the emergence of behaviors that are aided by the mechanical properties of a physical system.

1.3.4 Design of Soft Robots

Hardware design of soft robots generally consists of fluidic-based designs or hydraulic-based designs. We focus mainly on fluidic-based systems in this work. Systems such as (Katzschmann, Marchese, and Rus, Katzschmann et al.) utilized a lost-wax casting process to fabricate a planar soft manipulator gripper that was then used in an autonomous manipulation of diverse objects of different geometry within the soft manipulator's workspace. This manipulator had six bidirectional segments with cylindrical cavities forming an arm with a pleated shape. (Ilievski et al., 2011) developed a starfish-shaped soft robot that employed pneumatic networks within an elastomer to manipulate a living mouse without harming it. Soft lithography (Xia and Whitesides, 1998) is increasingly being used to design soft robots due to its capacity for replicating microstructures on a non-planar surface and the manner in which it provides access to three-dimensional structures. Soft lithography tolerates a wide range of materials and chemistry surfaces. Soft lithograph was used by (Shepherd et al., 2011) to fabricate a pneumatically-actuated robot that generated dexterous gaits and complex motions including crawling as well as undulation gaits. McKibben actuators and electro-active polymers are used for extension-based manipulation of objects (Pujana-Arrese et al., 2007).

Embodied intelligence, inspired from the behavior of natural organisms in nature, have inspired the prototype of a robot arm based on an artificial muscular hydrostat (Laschi et al., 2012). With tentacles made out of silicone conical arms, inspired from the muscular hydrostat of the Octopus vulgaris, the platform can change its shape around various geometrically-shaped objects. Bending fluidic actuators have garnered large attention in soft robotics owing to their robustness, continuous interaction with the environment, high force density, and large deformations. The FASTT (Cacucciolo et al., 2015) soft robot capitalizes on the compliance of bending fluidic actuators to realize a self-stabilizing locomotion on terrains with varying surface geometry. A recent class of elastomeric actuators are the fiber reinforced elastomeric enclosures (FREEs). These materials consists of cylindrical rubber or silicone pneumatic cavities that have fiber windings around them in specific

helical configurations to create sophisticated motions (such as compression, axial rotation, and twisting) when pressurized (Bishop-Moser et al., 2012; Demirkoparan and Pence, 2007; Holzapfel et al., 2000; Sedal et al., 2018). These actuators tend to be robust for linear extension, and twisting moments based manipulation.

1.3.5 Soft Actuators: Modeling and Control Approaches

Soft robots are notoriously difficult to control, given their continuum-based mechanical properties. Schemes for controlling these class of robots are a very active area of research with mixed successes. In what follows, we present a review of the modeling and control schemes that have erstwhile been employed in soft robotics. Approaches range from finite element modeling (Bern et al., 2017; Gent, 2012; Nesme et al., 2006, 2005), system identification and function-approximator based approaches (Giorelli et al., 2015; Ogunmolu et al., 2017), homogeneous continuum approaches (Holzapfel et al., 2000; Ogden, 1997), non-homogeneous continuum approaches (Ogden, 1997), to non-constant curvature approaches. Non-constant curvature approaches belong largely to three categories namely, the continuum approximation of hyper-redundant sytems such as found in (Chirikjian, 1994; Chirikjian and Burdick, 1995; Mochiyama, 2005), spring-mass models for semi-rigid robots such as (Yekutieli et al., 2005; Zheng et al., 2012), and geometric continuum models (Boyer et al., 2006; Demirkoparan and Pence, 2007; Gent, 2012; Holzapfel et al., 2000; Ogden, 1997; Rucker et al., 2010; Sedal et al., 2018). Cosserat approaches for semi-rigid soft robots, whereupon the strains of the Cosserat model are derived from stress-strain laws have been used by (Renda et al., 2014). In (Renda and Seneviratne, 2018), the authors derived a discrete Cosserat model for a soft-rigid multi-body system using the geometrical framework of the rigid robotics based on the exponential map.

For an extensive literature review of the design, fabrication, and control of continuum soft robots, we refer readers to (George Thuruthel et al., 2018; Rus, Daniela; Tolley, Michael T., 2015; Trivedi et al., 2008). Our goal in this dissertation is not the explicit design of soft robots per se but

the continuum modeling of individual soft robots, their kinematic configuration and dynamics in multi-DOF assembly modes and their control.

Aside from the need to develop the hardware consistent for a feedback control of patient position correction system, a separate *treatment planning optimization* problem is solved in order to assure a near-optimal treatment dose. Given an ideal dose description by a doctor as a biological statement of objectives, the *treatment planning optimization* problem uses *inverse planning* to generate a best set of intensity-modulated beams in a process often referred to as *beam orientation optimization*.

1.4 Beam Orientation Optimization

Radiation intensities are generally delivered from different beam orientations using *multiple collimators* – generating concave dose distributions that precisely spare sensitive normal organs with complex treatment geometries (Webb, 2001). Each beam is divided into beamlets, where beamlets in a beam are aimed from the same angle; it is noteworthy to add that a beamlet may be of a different intensity from that of its neighbor – helping generate the non-uniform radiation profiles on a target volume. The process of selecting the best combination of beam angles among the cardinality of beam orientations from which radiation could to be delivered is termed *beam orientation optimization* (BOO), while the process of optimizing the resulting beams so as to fit a doctor's prescribed dose by determining the intensity (influence) to assign each beamlet is termed *fluence map optimization* (FMO). Both of these are part of the so-called *inverse-planning* problem that uses optimization techniques to generate intensity distributions across a beam's eye-view and the geometric field shape of a target volume. When just the gantry of the LINAC machine is rotated with respect to the other angles of the robot, this results in a set of *coplanar beams* being swept out by the gantry.

Finding the optimum physically deliverable fluence profile to obtain a desirable dose is a key problem in IMRT. Mostly, this problem proceeds manually in a time-consuming scenario in most clinics. The success of this manual procedure depends on he treatment planner's experience. A

favorable radiation "influence" (fluence) profile is typically found by simultaneously maximizing and minimizing the dose delivered to tumors and OARs respectively, while generating sharp dose gradients at the transition between tumors and OARs. Since tumors often intersect with OARs, and the dose deposition's physics changes with every beam orientation, BOO is a non-convex problem (Craft, 2007), (Söderström and Brahme, 1993) with myriad possible beam combinations within a setup's phase space. One may consider the BOO problem as a set cover or combinatorial optimization search problem, where given a universe of all candidate beam angles, $\mathcal{U}$, we seek to find from a subset family $\mathcal{S}$ of $\mathcal{U}$, a cover subfamily, $\mathcal{C} \subseteq \mathcal{S}$ whose union is an optimal beam set that meets a doctor's prescription.

Techniques that have so far been proffered for numerically solving the BOO problem elongate the treatment planning process and lack adequate real-time feasibility. The reason is largely two-fold: first, a large, ill-conditioned dose influence matrix, $\mathcal{D}_{ij}$, for all discretized beam angles within the setup's phase continuum must be computed for every candidate beam angle in a very computationally demanding scheme; second, the respective beamlet's dose per angle must be used to solve the FMO solution for myriad beam combinations during search before the "best" beam angle set is chosen. $\mathcal{D}_{ij}(\theta_k)$ is interpreted as the dose delivered to a discretized voxel i by a unit intensity beamlet j, incident from angle θ_k, where k are the indices of beamlets in a beam B.

Linear programming duality theory alongside gradient descent have been used in computationally finding desirable beam angles and beamlet intensities (Bertsimas et al., 2013; Craft, 2007; Stein et al., 1997). These methods employ local search to evaluate the structure of an IMRT plan's objective function in order to improve a global plan quality. (Bertsimas et al., 2013) defined a linear programming problem with constraints that capture a doctor's preference for dose delivery: a gradient-based approach searched the neighborhood of angles earlier found from global beams search. In (Craft, 2007), the author found a beam angle set in the global beam angle space and then refined the angles locally with gradient descent optimization.

A popular computational approach is simulated annealing (SA), where gantry angles are sampled in the global beam space and corresponding beam profiles are optimized using local neighborhood

search. The local neighborhood search may consist of deterministic or stochastic search algorithms so as to improve a current solution. A fixed subset of a gantry angle's neighbors, or the complete neighborhood can be traversed at each iteration so as to find a better solution in the deterministic case, while in the stochastic case, SA arbitrarily chooses neighboring solutions so as to find an improved solution at each iteration (Aleman et al., 2008a). We refer readers to (Aleman et al., 2008a; Bortfeld and Schlegel, 1993; Pugachev et al., 2000; Pugachev and Xing, 2002; Stein et al., 1997) for BOO treatments using SA. We note that while SA may find a good solution, this usually comes at the expense of a long iteration of escaping local minima.

In approaches such as (Pugachev and Xing, 2001), a score function characterizes the maximum dose that a planning target volume (PTV) can receive from a beamlet, such that the dose tolerance of OARs and normal tissues along a beamlet's path are not exceeded. In (Djajaputra et al., 2003), the authors accelerated the IMRT optimization process using a fast SA approach to select beam angles from a set of predefined directions, while (Bortfeld and Schlegel, 1993) employed a frequency-domain paradigm for optimizing least-square-like objective functions using Parseval's theorem to the end of speeding up the simulation process. They reported being able to solve the nonconvex BOO problem within a "few minutes on a minicomputer". However, (Bortfeld and Schlegel, 1993)'s algorithm is restricted to l_2-like objective functions, and it assumes approximations on the mathematical model of the problem. Our approach in this work is adaptable to any family of objectives and solves the beam selection problem within the fraction of a minute.

Feature-based machine learning approaches are also used, whereupon the high dimensionality of the 3D treatment volume for a patient is divided into pencil beams such that an input-output mapping between the patients and the corresponding intensities are modeled using regression techniques such as support vector regression, or local regression (Lu et al., 2006). Such methods need a lot of training data to efficiently learn a good principal model that accurately maps the relationship between a patient's geometry and beam intensity. In (Li and Lei, 2010), DNA-computing and a genetic algorithm was implemented: beam angles were encoded with a DNA computing framework,

which generated genetic operators that were employed to search for feasible beam solutions. Other lines of work have treated IMRT treatment planning as an inverse optimization problem, with techniques ranging from adaptive l_{21} optimization (Jia et al., 2011), and mixed integer linear programming (D D'Souza et al., 2004; Lim et al., 2007; Wang et al., 2003).

1.5 Research Motivation

Being a relatively new form of cancer treatment, the full development of radiation therapy requires the interdisciplinary effort of engineers, physicists, and physicians in order to improve the treatment planning process. This is in part the purpose of this dissertation. Given the stringent requirements for accurate and precise delivery of dose for disease locations, as well as the elongated treatment planning process due to the heavy computational requirement of BOO, IMRT treatment planning require a sizable investment in clinical time from trained medical physicists, physicians, dosimetrists, and radiation therapists inter alia. The instrumentation and automation procedures in IMRT are in their infancy: from the 3D multimodal medical imaging of tumor sites to the robotic manipulation of patients on the treatment couch. More robustness and automation mechanisms are needed to improve the treatment of disease sites (Boyer et al., 2001). As such, the complete development of IMRT will require the multidisciplinary concerted effort of physicists, mathematicians, oncologists, roboticists, and broadly engineers.

We address the following concerns:

- How can we improving the time required for finding beam angles whilst preserving treatment quality?

- Can multi-DOF soft manipulator mechanisms function as positioning mechanisms in non-invasive radiosurgery such that radiation dose is not attenuated and motion-correction is comfortable compared with the rigid metallic masks that are used in most research clinics?

Given the inconvenience of frame- and mask-based immobilization devices currently used in IMRT, and the exposure of OARs to toxicity given the gyrations in patient motion as a result of respiratory motion, interfractional and intrafractional calibration errors (Sterzing et al., 2011), we synthesize, analyze and evaluate soft robot actuator position-correction mechanisms in non-invasive head and neck radiation therapy.

On a second note, we design an approximate policy iteration scheme (Bertsekas, 2013) for the time-exhausting IMRT inverse planning optimization process. Being a machine learning based model, we aim to avoid the manual feature selection for classifying the characterization of the mapping from patient geometries to beam orientation. Approximate policy iteration automation scheme has attractive *sub-optimal control* properties (Bertsekas et al., 2017a) when combined with recent function approximator breakthroughs in large state spaces (Levine et al., 2016; Mnih et al., 2015; Silver et al., 2016, 2017). Having a near-optimal automated tool for choosing beam angles will reduce the current trial and error process, and jump-start the beam angles set from which treatment planners could further fine-tune predicted beams given a patient's CT geometry in little time.

1.6 Dissertation Structure

The first part of this work treats the automation of the patient's motion correction system with arrangements of nonlinear elastic soft robot (in geometric primitive forms) around the patient's head and neck region. Some of this work have previously appeared in the following publications (Ogunmolu et al., 2015, 2017, 2015b, 2016b), and (Almubarak et al., 2018). Our additional contribution is the continuum mechanical model of the kinematics of strain deformation for a soft robot and a multi-dof analysis of motion of the head region under the influence of multiple soft robots to adjust infinitesimal patient motion on a treatment table. These can be found in Chapters 2 and 3.

In the second part of this work, we formally introduce the beam orientation optimization problem, and then prescribe an approximate dynamic programming approach that reduces the treatment

planning time for dosimetrists, radiation therapists and physicians during treatment. Some of the work described pertaining to this have appeared at radiation therapy conferences and algorithms for robotics venues e.g. (Sadeghnejad Barkousaraie et al., 2019a),(Sadeghnejad Barkousaraie et al., 2019b), (Barkousaraie et al., 2019), (Ogunmolu et al., 2018), (Sadeghnejad Barkousaraie et al., 2019), (Ogunmolu et al., 2019), (Ogunmolu et al., 2018). The new contributions here are the approximate policy iteration formulation which encompasses a column-generation supervised machine learning of correct beam angles (Bertsekas, 2013), which is in turn followed by a single agent Monte-Carlo tree search formulation. This is presented in Chapter 6.

In the course of my degree, I did some other work on embedding robustness into deep policies for robot navigation. These works are not discussed in this thesis but the reader can find them in the following publications: (Ogunmolu et al., 2018; Summers et al., 2017) and (Ogunmolu et al., 2017). All the codes for the experiments presented in this thesis as well as throughout my graduate degree are indexed on github.

CHAPTER 2

DEFORMATION ANALYSIS OF A SOFT CONTINUUM MANIPULATOR

In this chapter, we analyze a soft continuum manipulator molded in a spherical form using nonlinear elastic theory of deformations. Specifically, we abstract the system of motion interaction between multiple IABs and a representative head of a patient using the simple interaction between a single IAB and the head. We then write out the contact force equations of the head-neck-IAB system in order to analyze the forward kinematic equations of the patient's motion on a treatment table/machine. Unlike link lengths and joint angles used in parameterizing the motion of rigid robots, soft robots present unique challenges with respect to the nonlinearity of their elastic material properties, their potential infinite degrees of freedom, their morphological properties under strain and torsion (Sedal et al., 2018; Zambrano et al., 2014), and the lack of a well-developed framework for capturing their dynamical behavior and control (George Thuruthel et al., 2018).

In addition to the modeling methods described in § 1.3.5, we briefly describe our motivation for devising a finite elastic model for our robot mechanism. The constant curvature approach for parameterizing the deformation soft continuum robots (Hannan and Walker, 2000, 2003; Jones and Walker, 2006) has played significant role in the kinematic synthesis of deformable continuum models over the past two decades. Under this framework, it is assumed that the configuration space of a continuum module can be parameterized by three variables that relate the curvature of an arc projected on the soft robot's body to the length of the arc, and the angle subtended by a tangent along that arc. The relationship between these parameters are typically found using differential kinematic analysis with a Frenet-Serret frame that models the behavior of a curve on the soft robot's surface with or without torsion. By abstracting an infinite dimensional structure to 3D, large portions of the manipulator dynamics are discarded under the assumption that the actuator design is symmetric and uniform in shape. When multiple soft robots are arranged in a kinematic chain, the constant curvature sections can be stitched together to form the so-called piece-wise constant curvature model (Jones and Walker, 2006). For finite elastic deformations such

as continuum models made out of materials with Poisson ratio close to 0.5, the constant curvature model is an overly simplified model that exhibits poor performance in position control (Kapadia et al., 2014). It is noteworthy that the Cosserat brothers' beam theory has been relatively successful in modeling soft continuum dynamics (Renda et al., 2014; Trivedi et al., 2008). However, their heavy computational requirement given a more complex model and the added sensing cost does not scale well compared to simpler models. Finite element modeling methods, while accurate, require heavy computation and time-consuming calculations so that they are not very useful for our application.

Therefore, we rigorously analyze the deformation and kinematics of a soft continuum module built out of materials with incompressible walls, henceforth called *inflatable air bladders* (IAB), using nonlinear finite elastic deformation theory (Mooney, 1940; Ogden, 1997; Rivlin and Saunders, 1950; Treloar, 1975). This would enable us to build a broad hook for the model of the deformation of soft continuum robot modules under stress, strain, internal pressurization, and an arbitrary hydrostatic pressure. This model and kinematic decomposition will then be used to develop the kinematics of a 16 assembly mode (Merlet, 2015) soft robot motion correction mechanism. Foundational readings may be found in (Demirkoparan and Pence, 2007; Gent, 2012; Murray, 2017; Ogden, 1997; Treloar, 1975), but in the next section we introduce the basic principles.

2.1 Foundations

Let $\mathcal{E}$ be a set of elements, which we shall refer to as *points*. A set of points, a body B, describe *particles* that have a one-to-one mapping with a region, $\mathcal{B}$, of a Euclidean point space, $\mathcal{E}$. The body B is said to occupy $\mathcal{B}$. During motion, the region occupied by $\mathcal{B}$ in $\mathcal{E}$ continuously varies. The *configuration* of a body B is a mapping $\boldsymbol{\chi} : B \to \mathcal{E}$ which takes B's particles to their occupied locations in $\mathcal{E}$. The location occupied by a particle $X \in B$ in the configuration $\boldsymbol{\chi}$ is

$$\mathbf{x} = \boldsymbol{\chi}(X) \quad \Longrightarrow \quad X = \boldsymbol{\chi}^{-1}(\mathbf{x}). \tag{2.1}$$

We take $\boldsymbol{\chi}$ and $\boldsymbol{\chi}^{-1}$ to be $\mathcal{C}^2$-diffeomorphisms $i.e.$ they exist on twice continuously differentiable manifolds. The *motion* of B is parameterized by the configuration $\boldsymbol{\chi}_t : B \to \mathcal{E}$ for time $t \subset \mathbb{R}$, and can be written as

$$\mathbf{x} = \boldsymbol{\chi}_t(X) \quad \text{or} \quad \mathbf{x} = \boldsymbol{\chi}(X, t). \tag{2.2}$$

For a given particle, X, equation (2.2) describes a curve in $\mathcal{E}$, which is a path of X in the motion. A *time-independent* fixed configuration describes the *reference configuration* of a body B, and is written as

$$\boldsymbol{X} = \boldsymbol{\chi}_0(X), \quad X = \boldsymbol{\chi_0}^{-1}(\boldsymbol{X}), \tag{2.3}$$

where $\boldsymbol{X}$ denotes the location of the particle X in the reference configuration, $\boldsymbol{\chi}_0$. When a body deforms in the reference configuration, it enters a *current configuration*, which from (2.3) becomes,

$$\mathbf{x} = \boldsymbol{\chi}_t\{\boldsymbol{\chi}_0^{-1}(\boldsymbol{X})\}. \tag{2.4}$$

It therefore follows that $\mathbf{x} = \boldsymbol{\chi}_t(X)$. A motion in the current configuration $\mathbf{x} = \boldsymbol{\chi}_t(X)$ is rigid if and only if

$$\mathbf{x} = \boldsymbol{c(t)} + \boldsymbol{Q(t)X} \quad \forall \quad \boldsymbol{X} \in \mathcal{B}_0, \tag{2.5}$$

where the translatory motion of the body is encoded by $\boldsymbol{c}(t)$ while the pure rotatory motion of the body is defined by the proper second-order orthogonal tensor $\boldsymbol{Q}(t)$. We are concerned with the motion of a body from an initial configuration to a final configuration; we do not require the knowledge of transitory stages in the motion. Thus, abusing notation, we drop the t arguments and write the component form of (2.4) as

$$x_i = \boldsymbol{\chi}_i(\boldsymbol{X}), \quad i = 1, 2, 3. \tag{2.6}$$

with an arbitrary choice of reference configuration such that $\mathcal{B}_0 = \boldsymbol{\chi}_0(B)$. If a material body has all deformations for each material point in the body constrained to be *isochoric* (the material deforms

locally but volume is preserved globally), the material becomes *incompressible*, and the constraints condition is referred to as *incompressibility* (Ogden, 1997). The isochoric property is an important one, whereupon for finite stress-strain deformations and a class of large elastic deformations, it can model the physical property of the material in exact closed-form equations. Some results on isotropic finite elasticity have been laid out in the works of (Ogden, 1997; Rivlin and Saunders, 1950; Treloar, 1975) inter alia.

2.1.1 Group Theory Fundamentals

The special structure of a *group*, called the displacement group $\{D\}$, can be used to represent the motion of a soft body. The special orthogonal group $SO(3)$ is a subgroup of the general linear group

$$SO(3) = \{R \in GL(n, \mathbb{R}) : RR^T = \mathbf{I},\ \det R = \pm 1\}. \tag{2.7}$$

A group with a topology operation on its set of elements such that the group can be given the structure of a differential manifold with the property that group multiplication and inversion is continuous is called a *Lie group*. The special Euclidean matrix group $SE(3)$ is a differentiable manifold, comprised of all the translations and *proper* rotations that moves a body from one point to another in the ordinary cartesian 3-space $\mathbf{E}^3$ (Brockett, 1990). An isomorphism mapping maps open set of elements of $SE(3)$ onto an open set of $\mathbb{R}^6$. The special Euclidean group $SE(3)$ is composed of matrices of the form

$$g = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ \mathbf{0}^T & 1 \end{bmatrix} \in SE(3). \tag{2.8}$$

$\boldsymbol{R} \in SO(3)$ is a rotation matrix , and $\boldsymbol{p} \in \mathbb{R}^3$ represents the linear position of the particle. The *Lie algebra* is a vector space $\mathbf{L}$ with the antisymmetric bilinear operation $[,] : \mathbf{L} \times \mathbf{L} \to \mathbf{L}$ which satisfies the Jacobi identity, $[L_1, [L_2, L_3]] + [L_2, [L_3, L_1]][L_3, [L_1, L_2]] = 0$. The Lie algebra of $SE(3)$ is the vector space of all instantaneous velocities whose elements are 6-dimensional vectors

of the form (v, ω), where ω is the infinitesimal angular generator of the soft robot body, and v is the infinitesimal linear generator of the rigid body. It the representation

$$\mathfrak{se}(3) = \left\{ L | L = \begin{bmatrix} \hat{w} & v \\ 0 & 0 \end{bmatrix}, \hat{w} = -\hat{w}^T \right\} \tag{2.9}$$

where $\hat{w}$ is the skew-symmetric matrix,

$$\hat{\boldsymbol{w}} = \begin{bmatrix} 0 & -w_z & w_y \\ w_z & 0 & -w_x \\ -w_y & w_x & 0 \end{bmatrix} \in so(3). \tag{2.10}$$

The Lie algebra line element can be otherwise written as a 6-vector

$$\mathfrak{se}(3) = \begin{bmatrix} w^T & v^T \end{bmatrix}^T \in \mathbb{R}^6.$$

where $v, w \in \mathbb{R}^3$.

2.1.2 Screws and Twists

The twist is a screw and an amplitude, denoting an infinitesimal displacement. Forces and torques that produce velocity and acceleration are 3D vectors pairs, $(\mathbf{F}, \mathbf{M})$, called *wrench*. A twist and a wrench are reciprocal when $\omega.\mathbf{M} + v.\mathbf{F} = 0$. The screw is a geometrical object consisting of a line in space together with the *pitch*. The pitch is given by

$$h = \frac{w \cdot v}{w \cdot w}.$$

The twist, ξ, defined in generalized coordinates as a function of the body velocity, $\dot{g} \in \mathfrak{se}(3)$ is

$$\xi = \dot{g}\, g^{-1}, \tag{2.11}$$

from which the generalized velocity of a motion in world coordinates is defined as

$$\xi = \begin{bmatrix} \dot{R}R^T & \dot{p} - \dot{R}R^T p \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \hat{w} & v \\ 0 & 0 \end{bmatrix} \in \mathfrak{se}(3). \tag{2.12}$$

The isomorphism of the twist matrix in the Lie algebra to $\mathbb{R}^6$ can be represented as

$$\hat{\xi} = \begin{pmatrix} v^T & w^T \end{pmatrix}^T \tag{2.13}$$

2.1.3 The deformation gradient

Consider a reference configuration $\mathcal{B}_0$ of a body B, which deforms to a current configuration $\mathcal{B}_t$. From (2.5), the rectangular Cartesian coordinates component form become

$$x_i = \chi_i(X_\alpha). \tag{2.14}$$

From (2.14), we have the rate of deformation as

$$\mathrm{dx}_i = \frac{\partial x_i}{\partial X_\alpha}\mathrm{dX}_\alpha, \tag{2.15}$$

with invariant form

$$\mathbf{dx} = \mathbf{F}\mathbf{dX} \tag{2.16}$$

where $\mathbf{F}$ is a second-order tensor (using (Ogden, 1997)'s notation),

$$\mathbf{F} = \boldsymbol{\nabla} \otimes \boldsymbol{\chi}(\mathbf{X}), \tag{2.17}$$

and $\otimes$ denotes the outer product. For a *material line element* or *fiber* (a vector $\mathbf{dX}$) on a soft body, we require that $\mathbf{F}$ be non-singular since $\mathbf{F}\,\mathbf{dX} = 0$ is not physically realistic. Thus, $\mathbf{F}\mathrm{dX} \neq 0$ for all $\mathbf{dX} \neq 0$ so that $\det \mathbf{F} \neq 0$ is a restriction that is generally imposed on $\mathbf{F}$. We define the *left* and *right Cauchy-Green* deformation tensors as

$$\mathbf{B} = \mathbf{F}\,\mathbf{F}^T \quad \text{and} \quad \mathbf{C} = \mathbf{F}^T\mathbf{F}$$

respectively.

2.2 Deformation of a Single IAB

In this section, we present the spherical polar coordinate representation of IAB deformations, derive the deformation gradient under assumptions of *isochoric* deformations and *incompressibility* of the IAB material skin. We then parameterize the configuration space of IAB kinematics based on a relationship between internal pressure, *Cauchy stress*, stored strain energy, and the radii of the IAB. The IABs are made out of rubber or elastomeric components. These materials have the distinct property of incompressibility with a Poisson ratio of approximately 0.5 (Gent, 2012). Our overarching assumption is that volume does not change locally during deformation at $\mathbf{X}$ $i.e.$ the deformation obeys *isochoricity*.

2.2.1 Invariants of Deformation

A consistent mathematical theory that describes the deformation of highly elastic, incompressible, and isotropic materials under the action of applied forces was established by (Rivlin and Saunders, 1950) in terms of stored energy function, W. This stored strain energy, W, captures the physical properties of the material. It is a function of two invariants namely $I_1,$ and I_2, described in terms of the principal extension ratios of the strain ellipsoids, $\lambda_r, \lambda_\phi, \lambda_\theta$,

Stored Energy Invariants

$$I_1 = \lambda_r^2 + \lambda_\phi^2 + \lambda_\theta^2, \quad \text{and} \quad I_2 = \lambda_r^{-2} + \lambda_\phi^{-2} + \lambda_\theta^{-2}. \tag{2.18}$$

Under the incompressibility assumptions of the IAB material body, we have $\lambda_r\lambda_\phi\lambda_\theta = 1$ (Treloar, 1975). In spherical coordinates, the change in polar/azimuth angles as well as radii in the reference and current configurations are as illustrated in Figure 2.1. Forces that produce deformations can be derived using the strain energy-invariants relationship, particularly using the rates of change of the strain energy with respect to these invariants, $i.e.$ $\frac{\partial W}{\partial I_1}$ and $\frac{\partial W}{\partial I_2}$.

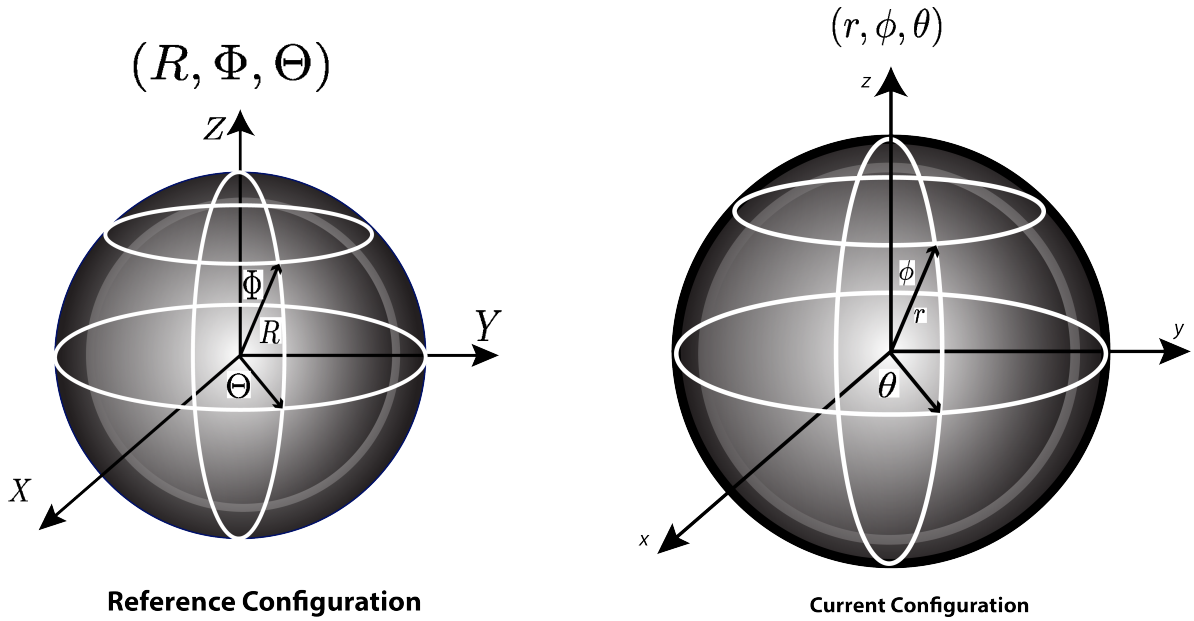


Figure 2.1: Deformation in spherical polar coordinates.

2.2.2 Analysis of Strain Deformations

Suppose a particle on the IAB material surface in the reference configuration has coordinates (R, Θ, Φ) defined in spherical polar coordinates, where R represents the radial distance of the particle from a fixed origin, Θ is the azimuth angle on a reference plane through the origin and orthogonal to the polar angle, Φ (see Figure 2.1). Denote the internal and external radii as $R_i,$ and R_o respectively. We define the following constraints,

$$R_i \leq R \leq R_o, \quad 0 \leq \Theta \leq 2\pi, \quad 0 \leq \Phi \leq \pi, \tag{2.19}$$

Now, suppose that the IAB undergoes deformation under the application of pressure to the internal walls of the tube as depicted in Figure 2.2. Arbitrary points A and A' in the reference configuration become Q and Q' in the current configuration. Suppose that the vector that describes the *material line element* that connects points A and A' is $a = a_R \boldsymbol{e}_r + a_\Theta \boldsymbol{e}_\Theta + a_\Phi \boldsymbol{e}_\Phi$ where $\boldsymbol{e}_R, \boldsymbol{e}_\Theta, \boldsymbol{e}_\Phi$ are respectively the basis vectors for polar directions R, Θ, Φ. We assume that there are internal constraints such that spherical symmetry is maintained during deformation of the incompressible material shell. This assumption helps in the simplification of the contact dynamics of the IAB

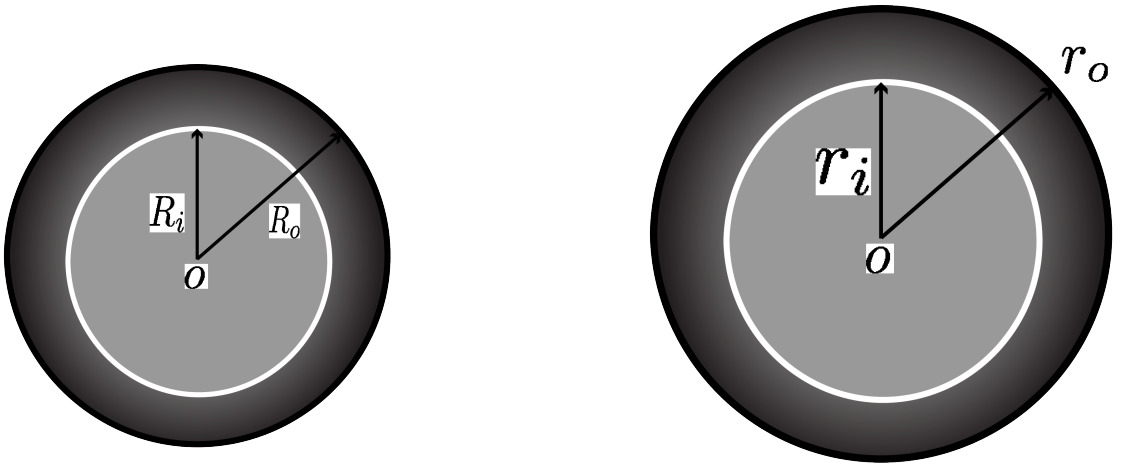


Figure 2.2: Radii change under deformation.

with the head as we shall see later on. More complicated deformation are left to a future work. With the spherical symmetry of the IAB preserved, we have the following constraints in the current configuration

$$r_i \leq r \leq r_o, \quad 0 \leq \theta \leq 2\pi, \quad 0 \leq \phi \leq \pi. \tag{2.20}$$

The radial vectors **R** and r are given in spherical coordinates

$$\mathbf{R} = \begin{bmatrix} R\,\cos\Theta\,\sin\Phi, \\ R\,\sin\Theta\,\sin\Phi, \\ R\,\cos\Phi \end{bmatrix} \quad \text{and} \quad \mathbf{r} = \begin{bmatrix} r\,\cos\theta\,\sin\phi, \\ r\,\sin\theta\,\sin\phi, \\ r\,\cos\phi \end{bmatrix}. \tag{2.21}$$

The material volume $\frac{4}{3}\pi\left(R^3 - R_i^3\right)$ contained between spherical shells of radii R and R_i remains constant throughout deformation, being equal in volume to $\frac{4}{3}\pi\left(r^3 - r_i^3\right)$ so that

$$\frac{4}{3}\pi\left(R^3 - R_i^3\right) = \frac{4}{3}\pi\left(r^3 - r_i^3\right) \quad \Longrightarrow \quad r^3 = R^3 + r_i^3 - R_i^3. \tag{2.22}$$

The homogeneous deformation between the two configurations imply that

$$r^3 = R^3 + r_i^3 - R_i^3, \quad \theta = \Theta, \quad \phi = \Phi, \tag{2.23}$$

where the coordinates obey the constraints of equations (2.19) and (2.20). Owing to the preservation of spherical symmetry, the *Lagrangean* and *Eulerian* axes coincide, with one axis aligned to the radial axis of the sphere and the other pair oriented arbitrarily normal to it so as to form a mutually orthogonal triad. The principal stretch along the azimuthal and zenith axes is therefore $\lambda = \lambda_\theta = \lambda_\phi = r/R$. Since for an isochoric deformation, $\lambda_r \cdot \lambda_\theta \cdot \lambda_\phi = 1$ (Treloar, 1975), it follows that the principal extension ratios are

$$\lambda_r = \frac{R^2}{r^2}; \lambda = \lambda_\theta = \lambda_\phi = \frac{r}{R},$$

where λ is the *azimuthal* and *polar principal stretch*. It is noteworthy that in the reference configuration, the principal stretches are $\{\lambda_i\}_{i=1,2,3} = 1$ such that $I_1, I_2, I_3 = 3$. The Mooney-Rivlin strain energy for small deformations as a function of the strain invariants of (2.18), is,

$$W' = C_1(I_1 - 3) + C_2(I_2 - 3). \tag{2.24}$$

where C_1 and C_2 are appropriate choices for the IAB material moduli. The Mooney form (2.24) has been shown to be valid even for large elastic deformations, provided that the elastic materials exhibit incompressibility and are isotropic in their reference configurations (Mooney, 1940). For mathematical scaling purposes that will soon become apparent, we rewrite (2.24) as $W = \frac{1}{2}W'$ so that

$$W = \frac{1}{2}C_1(I_1 - 3) + \frac{1}{2}C_2(I_2 - 3). \tag{2.25}$$

Note that equation(2.24) or (2.25) becomes the neo-Hookean strain energy relation when $C_2 = 0$. The deformation gradient $\mathbf{F}$ in spherical polar coordinates, may be written as

$$\begin{aligned} \mathbf{F} &= \lambda_r \boldsymbol{e}_r \otimes \boldsymbol{e}_R + \lambda_\phi \boldsymbol{e}_\phi \otimes \boldsymbol{e}_\Phi + \lambda_\theta \boldsymbol{e}_\theta \otimes \boldsymbol{e}_\Theta \\ &= \frac{R^2}{r^2}\boldsymbol{e}_r \otimes \boldsymbol{e}_R + \frac{r}{R}\boldsymbol{e}_\phi \otimes \boldsymbol{e}_\Phi + \frac{r}{R}\boldsymbol{e}_\theta \otimes \boldsymbol{e}_\Theta. \end{aligned} \tag{2.26}$$

The invariant equations, in polar coordinates, are therefore a function of the right Cauchy-Green, and finger (Kaye et al., 1998) deformation tensors given as

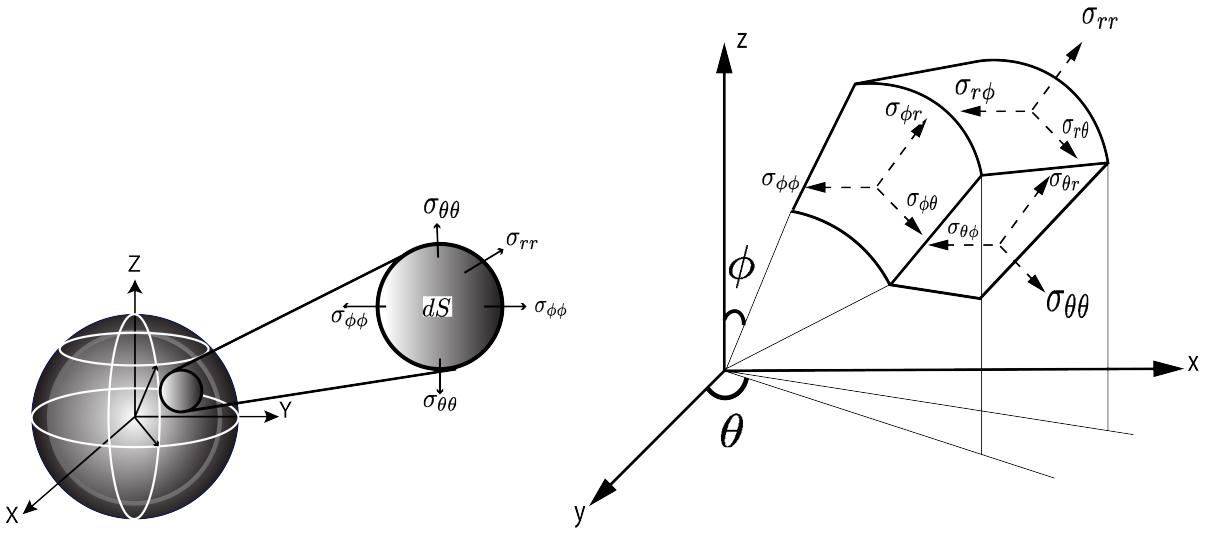


Figure 2.3: Body stress distribution on continuum's differential surface, dS.

IAB Invariants

$$I_1 = \mathbf{tr}(\mathbf{C}) = \frac{R^4}{r^4} + \frac{2\,r^2}{R^2} \tag{2.27a}$$

$$I_2 = \mathbf{tr}\left(\mathbf{C}^{-1}\right) = \frac{r^4}{R^4} + \frac{2\,R^2}{r^2}. \tag{2.27b}$$

2.2.3 Stress Laws and Constitutive Equations

At issue is the magnitudes of the differential stress in the IAB in spherical polar coordinates from a mechanical point of view; we will assume that thermodynamic properties such as temperature and entropy have little to no contribution. The IAB material stress response, **G** , at any point in the IAB at time t is instrumental in defining the Cauchy stress, $\boldsymbol{\sigma}$, and the history of the motion up to and including time t. The *constitutive equation* that relates the stress to an arbitrary motion will be determined using (Truesdell and Noll, 1965)'s *determinism for the stress principle*. The constitutive

relation for the nominal stress deformation for an elastic IAB material is given by

$$\boldsymbol{\sigma} = \mathbf{G}(\mathbf{F}) + q\mathbf{F}\frac{\partial \boldsymbol{\Lambda}}{\partial \mathbf{F}}(\mathbf{F}), \tag{2.28}$$

where $\mathbf{G}$ is a functional with respect to the configuration $\boldsymbol{\chi}_t$, q acts as a Lagrange multiplier, and $\boldsymbol{\Lambda}$ denotes the internal constraints of the IAB system. For an incompressible material, the indeterminate Lagrange multiplier becomes the hydrostatic pressure $i.e.\ q = -p$ (Holzapfel et al., 2000). The incompressibility isotropic assumption of the IAB material properties imply that $\boldsymbol{\Lambda} \equiv \det \mathbf{F} - 1$. As such, we find from (2.28) that

$$\begin{aligned} \boldsymbol{\sigma} &= \mathbf{G}(\mathbf{F}) - p\mathbf{F}\text{adj }^T(\mathbf{F}) \\ &= \mathbf{G}(\mathbf{F}) - p\mathbf{F}\mathbf{F}^{-T}\det(\mathbf{F}) \\ &= \mathbf{G}(\mathbf{F}) - p\mathbf{I} \end{aligned} \tag{2.29}$$

where the last part of the equation follows from the isochoricity, $\det(\mathbf{F}) = 1$. In terms of the stored strain energy, we can rewrite (2.29) as

$$\boldsymbol{\sigma} = \begin{bmatrix} \sigma_{rr} & \sigma_{r\phi} & \sigma_{r\theta} \\ \sigma_{\phi r} & \sigma_{\phi\phi} & \sigma_{\phi\theta} \\ \sigma_{\theta r} & \sigma_{\theta\phi} & \sigma_{\theta\theta} \end{bmatrix} = \frac{\partial W}{\partial \mathbf{F}}\mathbf{F}^T - p\mathbf{I}, \tag{2.30}$$

where $\mathbf{I}$ is the identity tensor and p represents an arbitrary hydrostatic pressure. A visualization of the component stresses of (2.30) on the walls of the IAB material is illustrated in Figure 2.3. It follows that

$$\begin{aligned} \boldsymbol{\sigma} &= \frac{\partial W}{\partial \mathrm{I}_1} \cdot \frac{\partial \mathrm{I}_1}{\partial \mathbf{F}}\mathbf{F}^T + \frac{\partial W}{\partial \mathrm{I}_2} \cdot \frac{\partial \mathrm{I}_2}{\partial \mathbf{F}}\mathbf{F}^T - p\mathbf{I} \\ &= \frac{1}{2}C_1\frac{\partial \mathrm{I}_1}{\partial \mathbf{F}}\mathbf{F}^T + \frac{1}{2}C_2\frac{\partial \mathrm{I}_2}{\partial \mathbf{F}}\mathbf{F}^T - p\mathbf{I} \\ &= \frac{1}{2}C_1\frac{\partial \mathbf{tr}\left(\mathbf{F}\mathbf{F}^T\right)}{\partial \mathbf{F}}\mathbf{F}^T + \frac{1}{2}C_2\frac{\partial \mathbf{tr}(\left[\mathbf{F}^T\,\mathbf{F}\right]^{-1})}{\partial \mathbf{F}}\mathbf{F}^T - p\,\mathbf{I} \\ &= C_1\mathbf{F}\mathbf{F}^T - C_2\left(\mathbf{F}^T\mathbf{F}\right)^{-2T} - p\mathbf{I} \\ &= C_1\mathbf{B} - C_2\mathbf{C}^{-2} - p\mathbf{I} \end{aligned} \tag{2.31}$$

where as before, $\mathbf{C} = \mathbf{F}^T\mathbf{F}$ and $\mathbf{B} = \mathbf{F}\mathbf{F}^T$. Thus, the constitutive law that governs the Cauchy stress tensor is

Stress-Strain Constitutive Law

$$\boldsymbol{\sigma} = C_1\mathbf{B} - C_2\mathbf{C}^{-2} - p\mathbf{I}. \tag{2.32}$$

It follows that the normal stresses are

$$\sigma_{rr} = -p + C_1\frac{R^4}{r^4} - C_2\frac{r^8}{R^8} \tag{2.33a}$$

$$\sigma_{\theta\theta} = \sigma_{\phi\phi} = -p + C_1\frac{r^2}{R^2} - C_2\frac{R^8}{r^8} \tag{2.33b}$$

2.3 Contact-Free IAB Boundary Value Problem

Here, for didactic purposes, we analyze the stress and internal pressure equations of the IAB at equilibrium. Later on, we describe the boundary value problem when the IAB is in contact with the head. Consider the IAB with boundary conditions given by,

$$\sigma_{rr}|_{R=R_o} = -P_{\text{atm}}, \quad \sigma_{rr}|_{R=R_i} = -P_{\text{atm}} - P \tag{2.34}$$

where P_{atm} is the atmospheric pressure (which may be set to 0) and $P > 0$ is the internal pressure exerted on the walls of the IAB above P_{atm} $i.e.$, $P > P_{\text{atm}}$. Suppose that the IAB stress components satisfy hydrostatic equilibrium, the equilibrium equations for the body force $\boldsymbol{b}'s$ physical component vectors, b_r, b_θ, b_ϕ are

$$-b_r = \frac{1}{r^2}\frac{\partial}{\partial r}(r^2\sigma_{rr}) + \frac{1}{r\sin\phi}\frac{\partial}{\partial\phi}(\sin\phi\sigma_{r\phi}) + \frac{1}{r\sin\phi}\frac{\partial}{\partial\theta}(\sigma_{r\theta}) - \frac{1}{r}(\sigma_{\theta\theta} + \sigma_{\phi\phi}) \tag{2.35a}$$

$$-b_\phi = \frac{1}{r^3}\frac{\partial}{\partial r}(r^3\sigma_{r\phi}) + \frac{1}{r\sin\phi}\frac{\partial}{\partial\phi}(\sin\phi\sigma_{\phi\phi}) + \frac{1}{r\sin\phi}\frac{\partial}{\partial\theta}(\sigma_{\theta\phi}) - \frac{\cot\phi}{r}(\sigma_{\theta\theta}) \tag{2.35b}$$

$$-b_\theta = \frac{1}{r^3}\frac{\partial}{\partial r}(r^3\sigma_{\theta r}) + \frac{1}{r\sin^2\phi}\frac{\partial}{\partial \phi}(\sin^2\phi\sigma_{\theta\phi}) + \frac{1}{r\sin\phi}\frac{\partial}{\partial\theta}(\sigma_{\theta\theta}) \tag{2.35c}$$

(see (Fung et al., 2001)). From the equation of balance of linear momentum (*Cauchy's first law of motion*), we have

$$\text{div}\,\boldsymbol{\sigma}^T + \rho\mathbf{b} = \rho\dot{\mathbf{v}} \tag{2.36}$$

where ρ is the IAB body mass density and $\mathbf{v}(\mathbf{x},t) = \dot{\boldsymbol{\chi}}_t(\boldsymbol{X})$ is the velocity gradient. Owing to the incompressibility assumption, we remark in passing that the mass density is uniform throughout the body of the IAB material. When the IAB is at rest, $\dot{\mathbf{v}}_t(\mathbf{x}) = 0\,\forall\,\mathbf{x} \in \mathcal{B}$ such that equation (2.36) loses its dependence on time. The assumed regularity of (2.1) thus leads to the steady state conditions for Cauchy's first equation; the stress field $\boldsymbol{\sigma}$ becomes *self-equilibrated* by virtue of the spatial divergence and the symmetric properties of the stress tensor, so that we have

$$\frac{1}{r}\frac{\partial}{\partial r}(r^2\sigma_{rr}) = (\sigma_{\theta\theta} + \sigma_{\phi\phi}). \tag{2.37}$$

Expanding, we find that

$$\begin{aligned}
\frac{1}{r}\left[r^2\frac{\partial\sigma_{rr}}{\partial r} + \sigma_{rr}\frac{\partial(r^2)}{\partial r}\right] &= (\sigma_{\theta\theta} + \sigma_{\phi\phi}) \\
\frac{1}{r}\left[r^2\frac{\partial\sigma_{rr}}{\partial r} + 2r\sigma_{rr}\right] &= (\sigma_{\theta\theta} + \sigma_{\phi\phi}) \\
r\frac{\partial\sigma_{rr}}{\partial r} &= \sigma_{\theta\theta} + \sigma_{\phi\phi} - 2\sigma_{rr} \\
\frac{\partial\sigma_{rr}}{\partial r} &= \frac{1}{r}(\sigma_{\theta\theta} + \sigma_{\phi\phi} - 2\sigma_{rr}).
\end{aligned} \tag{2.38}$$

Integrating the above equation in the variable r, taking $\sigma_{rr}(r_\circ) = 0$, and carrying out a change of variables from r to R, we find that

$$\begin{aligned}
\sigma_{rr}(\delta) &= -\int_\delta^{r_\circ}\frac{1}{r}(\sigma_{\theta\theta} + \sigma_{\phi\phi} - 2\sigma_{rr})dr,\ \ r_i \le \delta \le r_\circ \\
&= -\int_\Delta^{R_\circ}\frac{1}{r}(\sigma_{\theta\theta} + \sigma_{\phi\phi} - 2\sigma_{rr})\frac{dr}{dR}dR,\ \ R_i \le \Delta \le R_\circ
\end{aligned}$$

$$= -\int_{\Delta}^{R_\circ} \frac{R^2}{r^2}(\sigma_{\theta\theta} + \sigma_{\phi\phi} - 2\sigma_{rr})dR, \quad R_i \leq \Delta \leq R_\circ$$
$$= -\int_{\Delta}^{R_\circ} \left[2C_1\left(\frac{1}{r} - \frac{R^6}{r^7}\right) - 2C_2\left(\frac{R^6}{r^7} - \frac{r}{R^2}\right)\right] dR. \tag{2.39}$$

In the same vein, using the boundary condition of $(2.34)|_2$ and taking the ambient pressure $P_{\text{atm}} = 0$, we find that

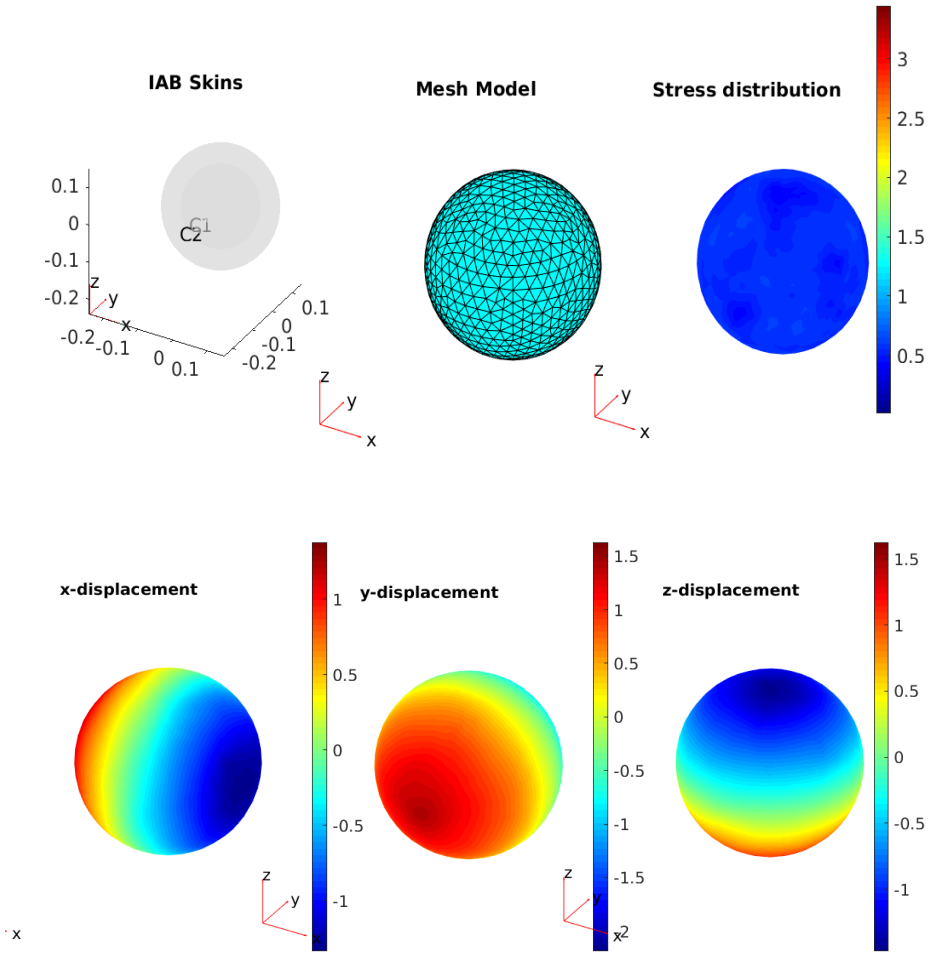
$$P = \int_{R_i}^{R_\circ} \frac{R^2}{r^2}(\sigma_{\theta\theta} + \sigma_{\phi\phi} - 2\sigma_{rr})dR, \quad R_i \leq R \leq R_\circ$$
$$= \int_{R_i}^{R_\circ} \frac{r^2}{R^2}\left[2C_1\left(\frac{1}{r} - \frac{R^6}{r^7}\right) - 2C_2\left(\frac{R^6}{r^7} - \frac{r}{R^2}\right)\right] dr$$
$$\equiv \int_{r_i}^{r_\circ} \left[2C_1\left(\frac{r^3}{R^4} - \frac{R^2}{r^3}\right) - 2C_2\left(\frac{R^4}{r^5} - \frac{r^3}{R^4}\right)\right] dr \tag{2.40}$$

where $r_i \leq r \leq r_\circ$. Equations (2.39) and (2.40) completely determine the deformation kinematics of the IAB material at rest. In the next chapter, we relate the head and neck force to the contact forces on the IAB surface boundary using the component stress laws just derived when the overall system is at rest.

2.3.1 Example: Radially symmetric deformation under gravity and applied internal pressure

For an isochoric deformation of a Cauchy-Elastic IAB material with material moduli and configuration radii as stated in the tables beneath the figures Figure 2.4, we would like to calculate and apply the pressure from (2.40) such that the soft robot radially deforms to $r_i = R_i + \delta cm$ for a given δ with a corresponding change in r_o as given by the relation $r_o = \sqrt[3]{R_\circ^3 + r_i^3 - R_i^3}$.

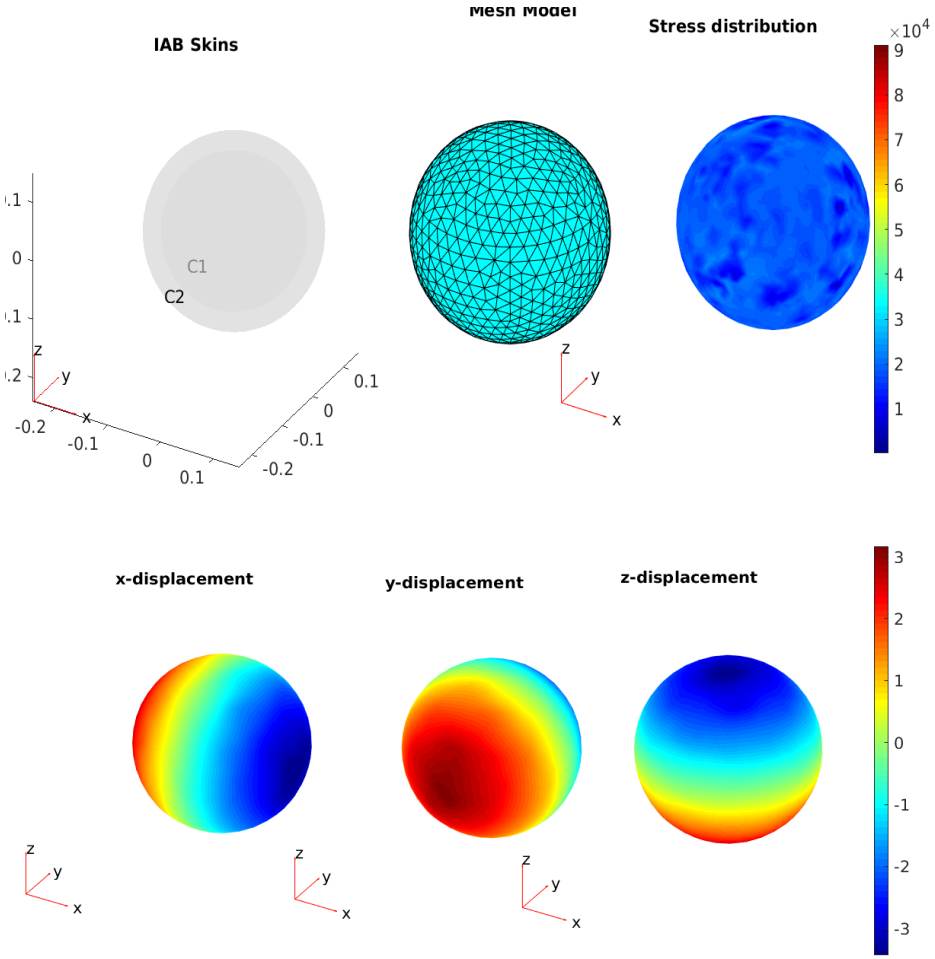
Figures 2.4, 2.5, 2.6, and 2.7 respectively illustrate the behavior of a representative volume of an isotropic and incompressible IAB material under the application of the derived internal pressures for a desired radial stretch. Whereas in 2.4 and 2.5, we are interested in a uniform expansion of the walls of the IAB material, in 2.6 and 2.7, we are concerned with a uniform contraction of the walls of the IAB skin. For each desired expansion of contraction in the figures, the calculated pressure is



C_1	C_2	$R_i(cm)$	$r_i(cm)$	$R_\circ(cm)$	$r_\circ(cm)$	P(psi)	ν	$T_{\text{mesh}}(s)$	$T_{\text{tot}}(s)$	$\rho(kg/m^3)$
$11,000$	$22,000$	10	13	15	16.60	14.552	0.45	0.8838	4.7782	9.8446×10^{-4}

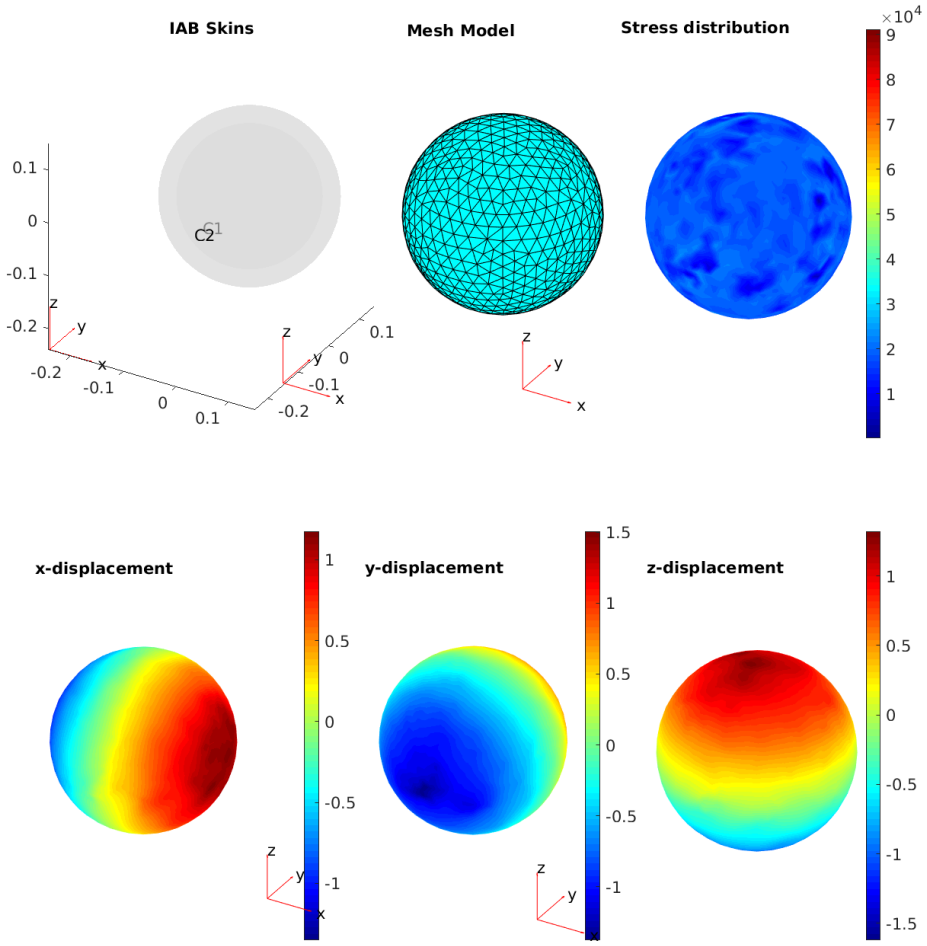
Figure 2.4: IAB Deformation (Extension)

as given in the tables beneath the figure. A negative pressure signifies air being pumped out of the bladder. In all, we notice a uniform displacement along the three Cartesian axes of the spherical bodies, demonstrating the utility of the derived kinematics for the IAB systems under hydrostatic equilibrium.



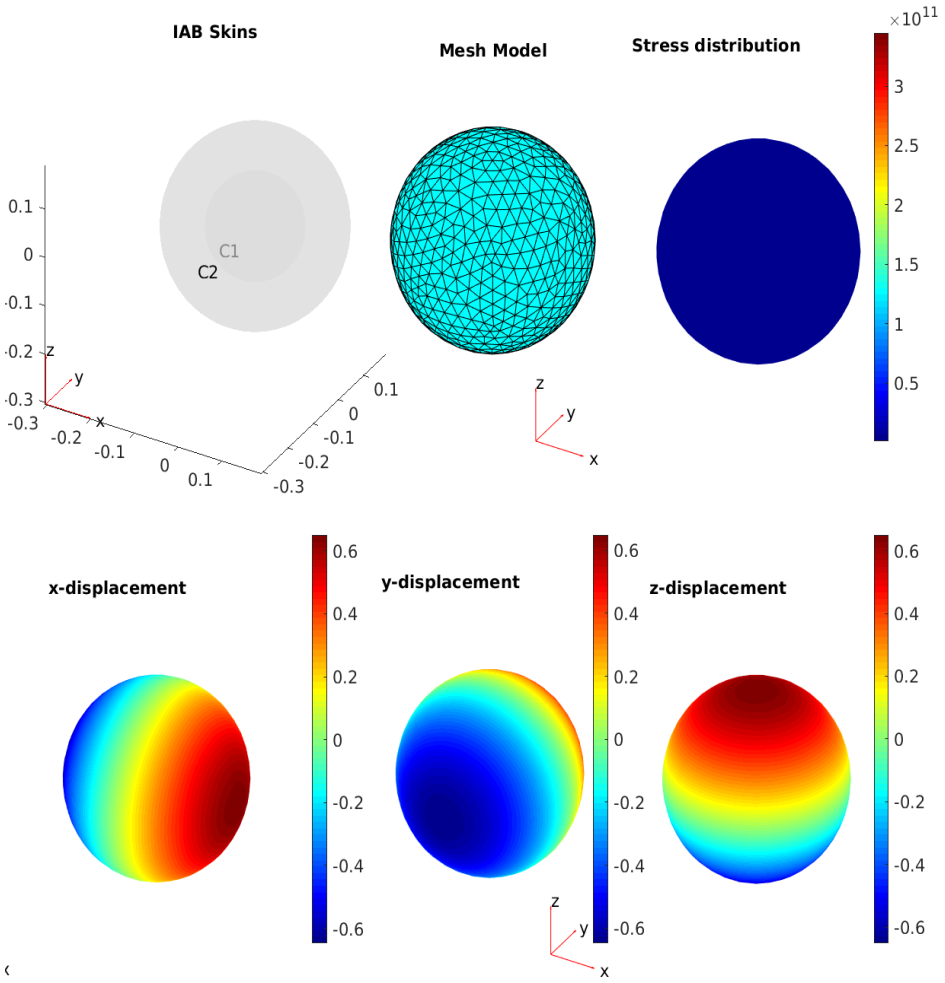
C_1	C_2	$R_i(cm)$	$r_i(cm)$	$R_\circ(cm)$	$r_\circ(cm)$	P(psi)	ν	$T_{\text{mesh}}(s)$	$T_{\text{tot}}(s)$	$\rho(kg/m^3)$
5e5	1e6	7.5	12	10	13.21	14.5193	0.4995	0.9143	4.1445	10^{-4}

Figure 2.5: IAB Deformation (Extension)



C_1	C_2	$R_i(cm)$	$r_i(cm)$	$R_\circ(cm)$	$r_\circ(cm)$	P(psi)	ν	$T_{\text{mesh}}(s)$	$T_{\text{tot}}(s)$	$\rho(kg/m^3)$
$500,000$	$1,200,000$	12	10	15	13.83	-27.3631	0.45	0.8625	4.5338	12×10^{-4}

Figure 2.6: IAB Deformation (Compression)



C_1	C_2	$R_i(cm)$	$r_i(cm)$	$R_\circ(cm)$	$r_\circ(cm)$	P(psi)	ν	$T_{\text{mesh}}(s)$	$T_{\text{tot}}(s)$	$\rho(kg/m^3)$
1.1×10^{12}	2.2×10^{10}	10	8	19	18.54	-27.3631	0.495	0.8236	4.5098	2×10^{-5}

Figure 2.7: IAB Deformation (Compression)

CHAPTER 3

KINEMATICS AND DYNAMICS OF A MULTI-DOF SOFT ROBOT

In this chapter, we will present systematic syntheses and analyses of parallel soft robot manipulator for head and neck immobilization. We will then derive the kinematics and dynamics necessary for motion planning and control of patients' head and neck on a treatment machine. Synthesizing multi-DOF parallel soft robots is challenging given the inter-dependency of the parameters that characterize the deformation, the individual robot constraints' relative three dimensional orientation, permitted motion orientations, the three dimensional relation between constraints and allowed motions, and the possibility of multiple assembly modes that may result in the same end-effector pose (Merlet, 2015). The configurations that we treat consists of soft robot links connected with extensible spring-like couplings; these spring-like couplings are chosen to exploit the soft structures' design for impedance control of the head and neck region of a patient. We analyze the manipulation map, kinematics and dynamics of the respective closed-loop chains, and analyze the contact equations between the IAB system and head.

3.1 A Soft Actuator Assembly for Patient Immobilization

Figure 3.1 shows the kinematic arrangement of the soft robots around the head. The geometry of this material has inflatable internal cavities that pressurize under the influence of fluids that flow via pneumatic hoses/tubes. These soft robot mechanisms consists of joints modeled by elastic springs to allow their adjustment around varying patient head sizes. The models are composed of three closed kinematic chains: one consisting of four soft actuators, connected to one another by 2-DOF spring-like flexible joints to allow the desirable range of motions. The other two chains are symmetric about the L-R plane (as described in Figure 1.6). The stiffness of these closed-loop kinematic chains is much higher than the common open-loop kinematic structures; deformation from passive compliance can be easily measured, the inherent elasticity of the rubber components

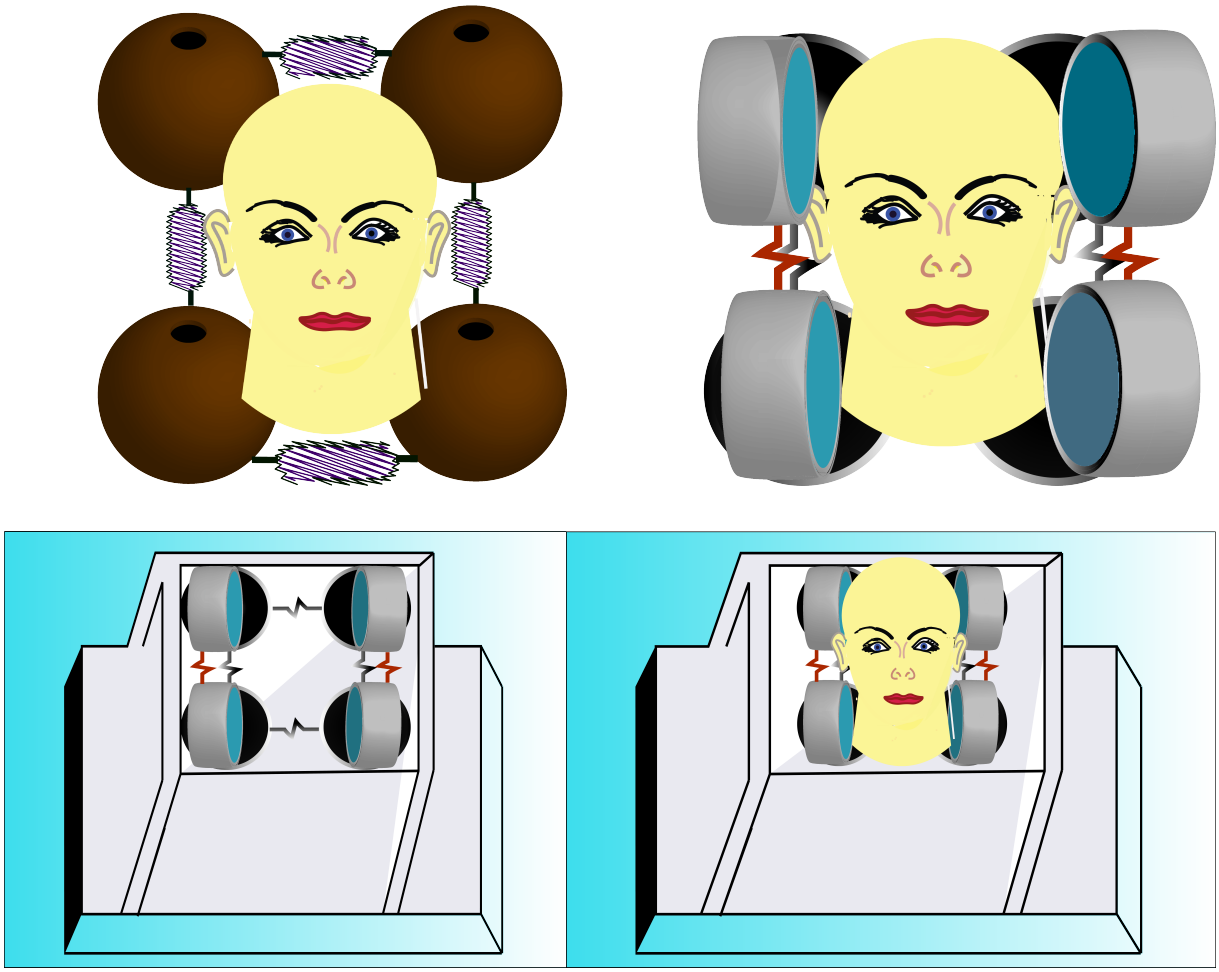


Figure 3.1: Soft Robots around Patient's Head and Neck Region on a Planar Table

increase passive compliance, and the controlled actuators can be used to generate a fixed behavior model.

The degrees of freedom of the configurations of Figure 3.1 can be determined using *Gruëbler-Kutzbach's mobility condition*, wherein the number of degrees of freedom of a mechanism is given by (when the actuation results in a non-planar workspace configuration)

$$F = 6(N - g) + \sum_{i=1}^{g} f_i \tag{3.1}$$

where N is the number of links in the mechanism, g is the number of joints, f_i is the total number of degrees of freedom for the ith joint. Equation (3.1) is only valid when the joints are independent. When the inflation of each soft robot in the configuration results in a planar workspace configuration, we have,

$$F = 3(N - g) + \sum_{i=1}^{g} f_i. \tag{3.2}$$

where through equations (3.1) and (3.2), the mechanism of Figure 3.1 have 16 DOFs since there are $N = 8$ links, $g = 8$ joints, and each joint has 2 DOFs .

3.2 Analysis of Contact Kinematics

We describe the contact between an IAB and the head through a mapping between the force exerted by the IAB at the contact point and the resultant forces at the center of mass of the head. We model the contact type between the head and an IAB similar to the soft finger contact primitive of (Nguyen, 1988). Here, our soft contact is the convex sum of *point contacts* with friction over the small area of contact. IAB forces and torques are modeled within a "cone of forces" about the direction of the surface normal from a patient's head (see Figure 3.2). The trajectory of the head under the influence of motion of an IAB is influenced by the position vector **r** of (2.21). When the IAB is pressurized, body forces over its current configuration and contact (*traction*) forces over its boundary $\partial\mathcal{B}$ impact an motion of the head. Constrained by the frictional coefficient, we define the soft contact force inside the friction cone as

$$\tilde{F}_{c_i} = \begin{bmatrix} \mathbf{I} & 0 \\ 0 & n_{c_i} \end{bmatrix} \begin{bmatrix} f_{c_i} \\ \tau_{c_i} \end{bmatrix}, \tag{3.3}$$

where $f_{c_i} \in \mathbb{R}^3$ denotes the amount of force exerted by the IAB along the direction of contact, $\tau_{c_i} \in \mathbb{R}$ is the moment of the contact force, and n_{c_i} is the *normal map* or *Gauss map*[1] for a manifold

[1]A normal map for a manifold S is a continuous map $g : S \to S^2 \subset \mathbb{R}^3$such that for every $s \in S, g(s)$ is orthogonal to S at s (Montana, 1988).

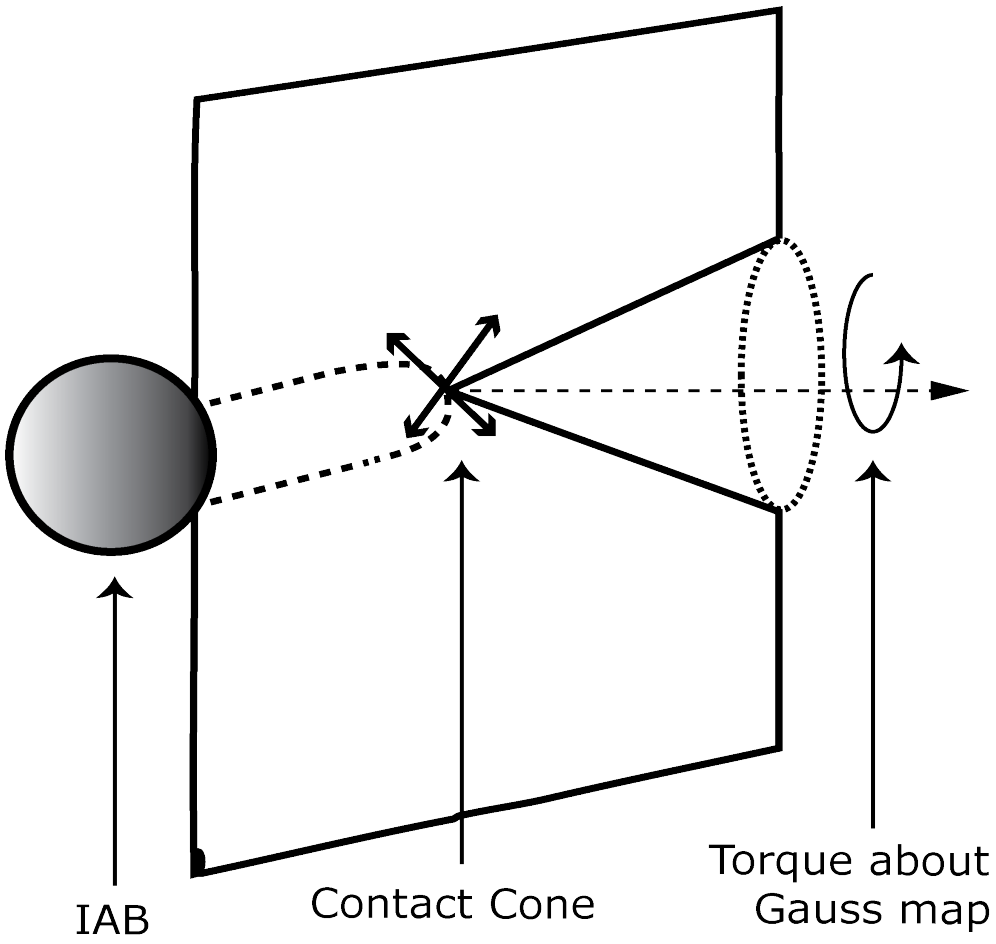


Figure 3.2: Illustration of the IAB Soft Contact Type

$S \subset \mathbb{R}^3$ of a head surface. For contact models with friction, we require that all contact forces lie within the friction cone, determined by the friction coefficient. The set of forces within or on the boundary of the friction cone is

$$FC = \{f_c \in \mathbb{R}^n : \|f^t_{c_{ij}}\| \leq \mu_{ij}\|f^n_{c_i}\|, \quad i = 1, \ldots, k, \quad j = 1, \ldots, m_i\} \tag{3.4}$$

where $f^t_{c_{ij}}$ denotes the tangent component of the j^{th} element of the contact force, $f^n_{c_i}$ is i^{th} contact's normal force, and μ_{ij} is $f_{c_{ij}}$'s coefficient of friction.

3.2.1 Contact-Based Boundary Value Problem for IAB

We now solve the boundary-value problem for the IAB deformation when in contact with the head as a follow-up to the analysis of § 2.3. Again, we assume a spherically-symmetric deformation constraint imposed on the IAB when the head rolls or slips on an IAB. This may be achieved through an appropriate vulcanization of the IAB rubber material, for example (see (Mooney, 1940) and (Gent, 2012)). We leave the analysis of spherical harmonics when the head dents the IAB such that the azimuthal symmetry of the IAB is not preserved to a future work. When the soft robots are in contact with the head, the applied forces on the current configuration of the IAB body $\mathcal{B}$ are

- the body forces, $\boldsymbol{b}$, of equation (2.35),
- the contact forces, f_c, at the IAB boundary, $\partial\mathcal{B}$, and
- the gravitational force of the head mass acting along the direction of contact, f_g.

We make the explicit assumption that the head maintains contact with the IAB throughout deformation. Suppose that for the i^{th} IAB, r_{c_i} represents the direction vector perpendicular from the point of contact to the center of the head cone of forces, it follows that the three equations that governs the motion of the IAB continuum are given as

$$\dot{\rho} + \rho \text{div}\mathbf{v} = 0 \tag{3.5a}$$

$$\boldsymbol{\sigma}^T = \boldsymbol{\sigma} \tag{3.5b}$$

$$\text{div}\boldsymbol{\sigma}^T + \rho\boldsymbol{b} = \rho\dot{\mathbf{v}} \tag{3.5c}$$

being respectively the conservation of mass (3.5a), the symmetry of the stress tensor (3.5b), and Cauchy's first law of motion (3.5c) respectively, (see (Ogden, 1997, pp. 150), for the derivation). In general, we expect that the mass of the body will be conserved given the incompressibility assumption of the IAB material, thus guaranteeing that (3.5a) is fulfilled. We have from (3.5c) and

the symmetric properties of the stress tensor that

$$\frac{1}{r^2}\frac{\partial}{\partial r}(r^2\sigma_{rr}) + \frac{1}{r\sin\phi}\frac{\partial}{\partial\phi}(\sin\phi\sigma_{r\phi}) + \frac{1}{r\sin\phi}\frac{\partial}{\partial\theta}(\sigma_{r\theta}) - \frac{1}{r}(\sigma_{\theta\theta}+\sigma_{\phi\phi}) + \rho b_r = \rho\ddot{\mathbf{r}}_x \tag{3.6a}$$

$$\frac{1}{r^3}\frac{\partial}{\partial r}(r^3\sigma_{r\phi}) + \frac{1}{r\sin\phi}\frac{\partial}{\partial\phi}(\sin\phi\sigma_{\phi\phi}) + \frac{1}{r\sin\phi}\frac{\partial}{\partial\theta}(\sigma_{\theta\phi}) - \frac{\cot\phi}{r}(\sigma_{\theta\theta}) + \rho b_\phi = \rho\ddot{\mathbf{r}}_y \tag{3.6b}$$

$$\frac{1}{r^3}\frac{\partial}{\partial r}(r^3\sigma_{\theta r}) + \frac{1}{r\sin^2\phi}\frac{\partial}{\partial\phi}(\sin^2\phi\sigma_{\theta\phi}) + \frac{1}{r\sin\phi}\frac{\partial}{\partial\theta}(\sigma_{\theta\theta}) + \rho b_\theta = \rho\ddot{\mathbf{r}}_z \tag{3.6c}$$

where $\ddot{\mathbf{r}}_x, \ddot{\mathbf{r}}_y$, and $\ddot{\mathbf{r}}_z$ are as defined in (B.2) and the body forces b_r, b_ϕ, b_θ are components of the gravitational force of the head acting on the IAB body $\boldsymbol{B}$. If the deformation is spherically symmetric, we expect that the shear stress component contributions $\sigma_{r\phi}, \sigma_{r\theta}, \sigma_{\phi\theta}$ would vanish in (3.6). It follows that the forces on the head (see derivation in Appendix C) are in part

$$\begin{aligned} P = \int_{r_i}^{r_\circ} \Bigg[\frac{1}{r}\left(-2p + 2C_1\frac{r^2}{R^2} - 2C_2\frac{R^8}{r^8}\right) - \rho b_r + \rho\cos\theta\left(2\dot{r}\dot{\phi}\cos\theta + r\cos\theta\ddot{\phi} - 2r\dot{\theta}\dot{\phi}\sin\theta\right) \\ -\rho\sin\phi\left(\cos\theta(-\ddot{r} + r\dot{\theta}^2 + r\dot{\phi}^2) + \sin\theta(2\dot{r}\dot{\theta} + r\ddot{\theta})\right)\Bigg]\, dr \end{aligned} \tag{3.7a}$$

$$\begin{aligned} \sigma_{\phi\phi}(\epsilon) = -\int_\epsilon^\pi \Big[r\rho\Big[\cos\phi\left(2r\dot{\theta}\dot{\phi}\cos\theta + (2\dot{r}\dot{\phi} + r\ddot{\phi})\sin\theta\right) + \\ \sin\theta\left(2\dot{r}\dot{\theta}\cos\theta + r\ddot{\theta}\cos\theta + (\ddot{r} - r\dot{\theta}^2 - r\dot{\phi}^2)\right)\sin\phi\Big] - \rho r b_\theta\Big]\, d\phi,\ 0 \le \epsilon \le \pi \end{aligned} \tag{3.7b}$$

$$\sigma_{\theta\theta}(\zeta) = -\int_\zeta^{2\pi}\left[-r\rho b_\theta\sin\phi + r\rho\sin\phi\cos\phi\left(\ddot{r} - r\dot{\phi}^2\right) - r\rho\sin^2\phi\left(2\dot{r}\dot{\phi} + r\ddot{\phi}\right)\right] d\theta,\ 0 \le \zeta \le 2\pi, \tag{3.7c}$$

the internal pressurization, and component stresses $\{P_i, \sigma_{\phi\phi}(\epsilon), \sigma_{\theta\theta}(\zeta)\}$, where $0 \le \epsilon \le \pi, 0 \le \zeta \le 2\pi$.

3.2.2 Contact Forces, IAB Stress Components, and Head Gravitational Force

We assume that the stress vector $\boldsymbol{\sigma}$ at a point on the IAB surface is uniform and continuous throughout the IAB boundary so that it linearly depends on the normal map (this follows from Cauchy's theorem; readers may see the proof in (Ogden, 1997, §3.3.1)). Recall that the correspondence between material line elements in the reference and current configuration is

$$\mathbf{dx = F\, dX} \quad \Longrightarrow \quad \mathbf{F}^{-T}\mathbf{dx = dX}.$$

Let $\mathbf{H} = \mathbf{F}^{-T}$ and $\mathbf{dA}$ represent an infinitesimal vector element on the material surface at a neighborhood of point $\mathbf{X}$ in $\mathcal{B}$ such that $\mathbf{dA} = \mathbf{N}$dA, where $\mathbf{N}$ is the unit outward normal to the IAB boundary $\partial\mathcal{B}_\circ$ in the reference configuration. The corresponding deformed surface of the IAB with normal $\mathbf{n}$ from a surface, da, of the IAB in the current configuration is $\mathbf{da} = \mathbf{n}\,$da. Using *Nanson's formula*, we have the following relation between surfaces in the reference and current configuration

$$\mathbf{da} = J\,\mathbf{H}\,\mathbf{dA} \qquad \Longrightarrow \qquad \mathbf{n}\,\text{da} = J\,\mathbf{H}\,\mathbf{N}\,\text{dA}. \tag{3.8}$$

where $J = \det \mathbf{F}$. Multiplying throughout equation (3.8) by the stress tensor of (2.32), the resultant contact force on the boundary $\partial\mathcal{B}$ in the current configuration may be written as (owing to the volume preservation on the boundary of the IAB material)

$$\int_{\partial\mathcal{B}} \boldsymbol{\sigma}\,\mathbf{n}\,\text{da} = \int_{\partial\mathcal{B}_\circ} J\,\boldsymbol{\sigma}\,\mathbf{H}\,\mathbf{N}\,\text{dA}. \tag{3.9}$$

The *Piola-Kirchoff* stress tensor field is defined as

$$\mathbf{S} = J\,\mathbf{H}^T\,\boldsymbol{\sigma}. \tag{3.10}$$

(see (Ogden, 1997, §4.2)). It follows that the force on an element surface da of the IAB in a configuration $\mathcal{B}$ is

$$\boldsymbol{\sigma}\mathbf{da} = \mathbf{S}^T\mathbf{dA}.$$

Thus, the contact force f_{c_i} on the boundary $\partial\mathcal{B}$ of the i^{th} IAB in a configuration $\mathcal{B}$ (as in (3.3)) is

$$f_{c_i} = \mathbf{S}_i^T\mathbf{dA}_i = J_i\boldsymbol{\sigma}_i\mathbf{H}_i\mathbf{dA}_i = J_i\boldsymbol{\sigma}_i\mathbf{F}_i^{-1}\mathbf{dA}_i \tag{3.11}$$

where (3.11) follows from the symmetric property of $\mathbf{F}_i$ and $\boldsymbol{\sigma}_i$. For the i^{th} IAB, at the region of contact, we have the contact force as

$$f_{c_i} = J_i\left(\frac{R_i^2}{r_i^2}P_i + \frac{R_i}{r_i}\sigma_{\phi\phi_i}(\epsilon) + \frac{R_i}{r_i}\sigma_{\theta\theta_i}(\zeta)\right)\mathbf{dA}_i \tag{3.12}$$

where $\sigma_{jj_i}(v)$ are the definite integrals of (3.7). Owing to the isochoric deformation assumption, we have from (3.12) that

$$f_{c_i} = \left(\frac{R_i^2}{r_i^2} P_i + \frac{R_i}{r_i} \sigma_{\phi\phi_i}(\epsilon) + \frac{R_i}{r_i} \sigma_{\theta\theta_i}(\zeta) \right) n_{c_i} dA_i. \tag{3.13}$$

where we have set the outward normal map $\mathbf{N}$ to n_{c_i} of (3.3). The torque is the moment of the contact force on the i^{th} IAB,and it is given by

$$\boldsymbol{\tau}_{c_i} = f_{c_i} \times r_{c_i} \tag{3.14}$$

where $r_{c_i} \in \mathbb{R}^3$ is the unit vector between the head reference point and the contact. The soft contact force of equation (3.3) can be re-stated in terms of the derived stress tensor of (2.32), the deformation gradient of (2.26) and the Piola-Kirchoff stress field of (3.10) $i.e.$

Friction Cones' Contact Force

$$\tilde{F}_{c_i} = \begin{bmatrix} \mathbf{I} & 0 \\ 0 & n_{c_i} \end{bmatrix} \begin{bmatrix} f_{c_i} \\ f_{c_i} \times r_{c_i} \end{bmatrix}. \tag{3.15}$$

where f_{c_i} and $\boldsymbol{\tau}_{c_i}$ are as given in equations (3.13) and (3.14).

3.2.3 Contact Coordinates and Head Velocity

The head will make contact with the IAB at multiple points on its surface, so we describe the kinematics of these contact points using an atlas[2] of contact coordinate charts. In this sentiment, let C_{r_1} and C_{r_h} respectively represent a fixed reference frame with respect to the IAB and head, H (see Figure 3.3). Furthermore, let $S_1 \subset \mathbb{R}^3$ and $S_h \subset \mathbb{R}^3$ denote the respective *orientable manifold*[3] embeddings of the IAB and head surfaces with respect to frames C_{r_1} and C_{r_h}. We

[2]An atlas $\tilde{S}$ is a set of surfaces where each surface $S \in \tilde{S}$ has an invertible map $f(\boldsymbol{u})$ from an open subset U of $\mathbb{R}^2$ to a surface $S \subset \mathbb{R}^3$ such that the partial derivatives $\frac{\partial f}{\partial u}(\boldsymbol{u})$, $\frac{\partial f}{\partial v}(\boldsymbol{v})$ are linearly independent for all $\boldsymbol{u} = (u, v) \in U$.

[3]An orientable manifold is a manifold S for which the Gauss map exists.

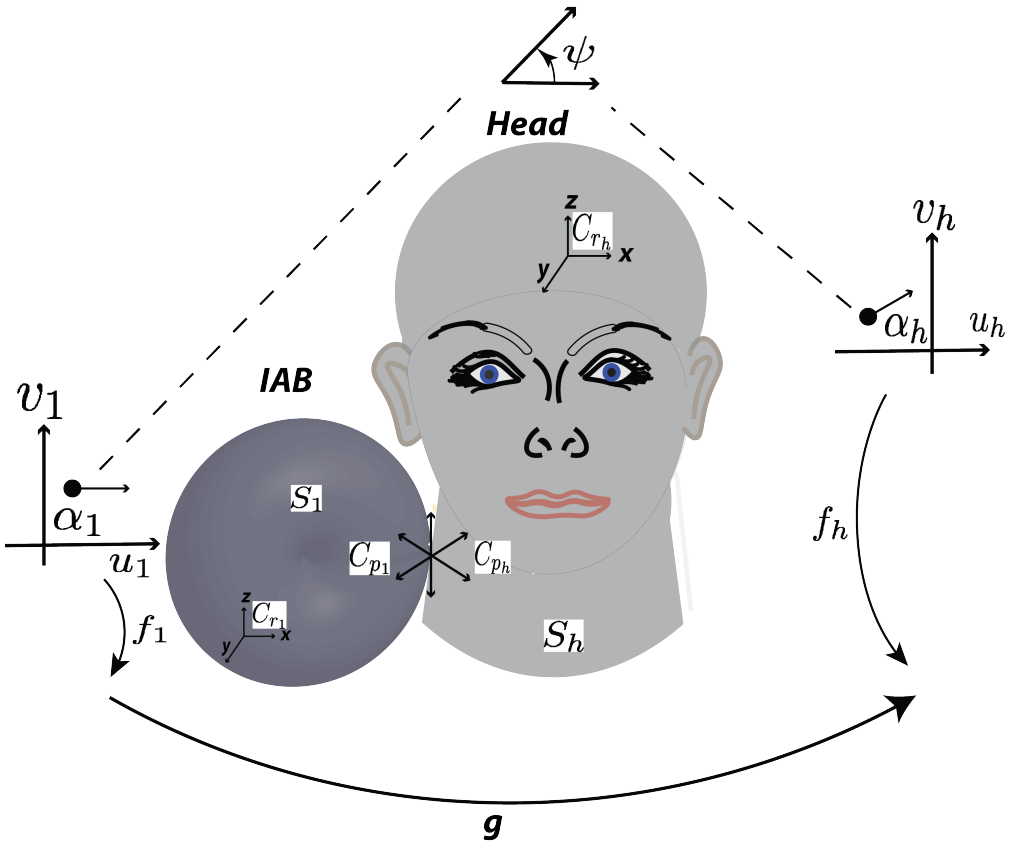


Figure 3.3: Sliding and rolling contact illustration of a single IAB and the Head

shall let S_1 and S_r belong to the *atlases* $\{S_{1_i}\}_{i=1}^{n_1}$, $\{S_{h_i}\}_{i=1}^{n_r}$ respectively. Suppose (f_1, U_1) and (f_r, U_r) are *coordinate systems* for the IAB and the head respectively, where f_i is an invertible map, $f_i(u_i, v_i) : U \to S_i \subset \mathbb{R}^3$

$$f_i(u_i, v_i) : \{U \to S_i \subset \mathbb{R}^3 | i = 1, h\},$$

from an open subset U of $\mathbb{R}^2$ to a *coordinate patch* $S_i \subset \mathbb{R}^3$ such that the partial derivatives $\frac{\partial f_i}{\partial u_i}$ and $\frac{\partial f_i}{\partial v_i}$ are linearly independent. Let $p_1(t) \in S_1$ and $p_h(t) \in S_h$ represent the positions of the contact points with respect to frames C_{r_1} and C_{r_h} respectively at time t. In general, the contact points $p_1(t)$ and $p_h(t)$ will not remain in the coordinate systems S_1 and S_h for all time. Thus, we choose an interval I where $p_1(t) \in S_{1_i}$ and $p_h(t) \in S_{h_j}$ for all $t \in I$ and some i and j. As seen in Figure 3.3,

C_{p_1} and C_{p_h} denote the contact frames that coincide with the *normalized Gauss frames* at p_1 and p_h for all $t \in I$, and $\alpha_1,\ \alpha_h$ are local coordinate frames that describe the IAB motion with respect to the head such that

$$\alpha_1 = (u_1, v_1) \in U_1, \text{ and } \alpha_h = (u_h, v_h) \in U_h.$$

Let the angle between the tangent planes of $\alpha_1,$ and α_h be ψ. The transformation matrix $g \in \Omega \subset$ *SE*(3) encodes the relative orientation and position of the IAB with respect to the head where Ω is the set of all relative positions and orientations in the atlasses $\{S_{1_i}\}_{i=1}^{n_1}$, $\{S_{h_i}\}_{i=1}^{n_h}$ for which the IAB and head remain in contact. We let the *contact coordinates* be described by $\eta = (\alpha_1, \alpha_h, \psi)$. The head's motion is governed by traction forces arising from the friction tangential to the IAB surface and the pressure normal to the IAB surface. Thus, at the points of contact, if $R \in SO(3)$ is the rotatory component of g, η must satisfy

$$g \circ f_1(\alpha_1) = f_h(\alpha_h) \tag{3.16a}$$

$$R\, n_1(\alpha_1) = -n_h(\alpha_h) \tag{3.16b}$$

since the contact locations must coincide for the IAB and the head, and the tangent planes must coincide so that the outward normal maps $n_1 : S_1 \to S^2 \subset \mathbb{R}^3$ and $n_h : S_h \to S^2 \subset \mathbb{R}^3$ agree. Furthermore, the orientation of the tangent planes of α_1 and α_h is the unique angle $\psi \in [0, 2\pi)$ between the x-axes of C_{p_1} and C_{p_h} such that

$$R\frac{\partial f_1}{\partial \alpha_1} M_1^{-1} R_\psi = \frac{\partial f_h}{\partial \alpha_h} M_h^{-1} \tag{3.17}$$

where M_i is a 2×2 square root of the Riemannian metric tensor (Spivak, 1979) that normalizes the columns of $\frac{\partial f}{\partial \alpha}$, $i.e.$

$$M_i = \begin{bmatrix} \|\frac{\partial f_i}{\partial u_i}\| & 0 \\ 0 & \|\frac{\partial f_h}{\partial v_i}\| \end{bmatrix} \tag{3.18}$$

and R_ψ is chosen such that a rotation of C_{p_1} about its z-axis through $-\psi$ radians aligns the x-axes of the local coordinate system α_1 to that of the head's local coordinate system α_h $i.e.$

$$R_\psi = \begin{bmatrix} \cos\psi & -\sin\psi \\ -\sin\psi & -\cos\psi \end{bmatrix}. \tag{3.19}$$

Notice that $R_\psi = R_\psi^T = R_\psi^{-1}$. We define the normalized Gauss frame at a point u on the surface U of the orthogonal coordinate system (f, U) as,

$$\begin{bmatrix} x_u & y_u & z_u \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial u}/\|\frac{\partial f}{\partial u}\| & \frac{\partial f}{\partial v}/\|\frac{\partial f}{\partial v}\| & n_u(f(u)) \end{bmatrix} \tag{3.20}$$

where x_u, y_u, and z_u are functions mapping $U \subset \mathbb{R}^2 \to \mathbb{R}^3$ and n_u is the continuous Gauss map $n_u : S \to S^2 \subset \mathbb{R}^3$. The motion of the contacts $\dot{\eta}$ as a function of components of the twist vector $\hat{\xi} = (v, w)^T$ is given in (3.21) as the respective *first, second*, and *third equations of contact*. Our derivation, which closely follows (Murray and Sastry, 1990) multi-fingered kinematics' proof, may be found in Appendix A.

Equations of Contact

$$\dot{\alpha}_h = M_h^{-1}(\mathcal{K}_h + \tilde{\mathcal{K}}_1)^{-1}\left(\omega_t - \tilde{\mathcal{K}}_1 v_t\right) \tag{3.21a}$$

$$\dot{\alpha}_1 = M_1^{-1} R_\psi (\mathcal{K}_h + \tilde{\mathcal{K}}_1)^{-1}(\omega_t - \mathcal{K}_h v_t) \tag{3.21b}$$

$$\dot{\psi} = \omega_n + T_h M_h \dot{\alpha}_h + T_1 M_1 \dot{\alpha}_1 \tag{3.21c}$$

where

$$T_h = y_h^T \frac{\partial x_h}{\partial \alpha_h} M_h^{-1}, \qquad T_1 = y_1^T \frac{\partial x_1}{\partial \alpha_1} M_1^{-1}, \qquad \omega_n = z_h^T \omega \tag{3.22a}$$

$$\mathcal{K}_h = \begin{bmatrix} x_h^T, & y_h^T \end{bmatrix}^T \frac{\partial n_h^T}{\partial \alpha_h} M_h^{-1}, \qquad \mathcal{K}_1 = R_\psi \begin{bmatrix} x_1^T, & y_1^T \end{bmatrix}^T \frac{\partial n_1^T}{\partial \alpha_1} M_1^{-1} R_\psi \tag{3.22b}$$

$$\omega_t = \begin{bmatrix} x_h^T, & y_h^T \end{bmatrix}^T \begin{bmatrix} n_h \times \omega \end{bmatrix}^T, \qquad v_t = \begin{bmatrix} x_h^T, & y_h^T \end{bmatrix}^T \begin{bmatrix} (-f_h \times \omega + v) \end{bmatrix}^T. \tag{3.22c}$$

Note that ω_t is the rolling velocity of the head projected onto the tangent plane of the contact and v_t is the sliding velocity; ω_n is the relative rotational velocity projected to the contact's surface normal, and $\tilde{\mathcal{K}}_1 = R_\psi \mathcal{K}_1 R_\psi$ is the curvature of the IAB with respect to the contact frame that coincides with the normalized Gauss frame at $p_1(t)$. The matrix $(\mathcal{K}_h + \tilde{\mathcal{K}}_1)^{-1}$ is the so-called *relative curvature* originally coined by (Montana, 1988). Simplifying equations (3.22), we find that

$$x_h = \frac{\partial f}{\partial u_h} / \| \frac{\partial f}{\partial u_h} \|, \quad y_h = \frac{\partial f}{\partial v_h} / \| \frac{\partial f}{\partial v_h} \|, \qquad z_h = n_u(f(u))$$

$$T_h = y_h \left[\tfrac{\partial x_h^T}{\partial u_h} / \| \tfrac{\partial f}{\partial u_h} \|, \tfrac{\partial x_h^T}{\partial v_h} / \| \tfrac{\partial f}{\partial v_h} \| \right], \qquad T_1 = y_1 \left[\tfrac{\partial x_1^T}{\partial u_1} / \| \tfrac{\partial f}{\partial u_1} \|, \tfrac{\partial x_1^T}{\partial v_1} / \| \tfrac{\partial f}{\partial v_1} \| \right]$$

$$\mathcal{K}_h = \begin{bmatrix} x_h^T, & y_h^T \end{bmatrix}^T \left[\tfrac{\partial n_h^T}{\partial u_h} / \| \tfrac{\partial f}{\partial u_h} \|, \tfrac{\partial n_h^T}{\partial v_h} / \| \tfrac{\partial f}{\partial v_h} \| \right], \qquad \mathcal{K}_1 = \begin{bmatrix} x_1^T, & y_1^T \end{bmatrix}^T \left[\tfrac{\partial n_1^T}{\partial u_1} / \| \tfrac{\partial f}{\partial u_1} \|, \tfrac{\partial n_1^T}{\partial v_1} / \| \tfrac{\partial f}{\partial v_1} \| \right]$$

We see that for the contact interaction between an IAB and the head, for a $U \subset \mathbb{R}^2$ we must choose an appropriate $f_i : U_i \to S_i \subset \mathbb{R}^3$ in order to characterize the setup.

3.2.4 Contact Kinematics Examples

On a spherical surface of an IAB with the following coordinate system,

$$U = \{(u, v) : -\pi/2 \leq u \leq \pi/2, \, -\pi \leq v \leq \pi\}, \quad f : U \to \mathbb{R}^3$$
$$f(u, v) \implies (R \cos u \cos v, -R \cos u \sin v, R \sin u) \tag{3.23}$$

for some radius R of the sphere. We choose the normal $n_u = 1/R\, f$. We find that the normalized Gauss coordinate is

$$x_h = \begin{bmatrix} -\sin(u)\, \cos(v) \\ \sin(u)\, \sin(v) \\ \cos(u) \end{bmatrix} \quad y_h = \begin{bmatrix} -\sin(v) \\ -\cos(v) \\ 0 \end{bmatrix} \quad z_h = \begin{bmatrix} \cos u \cos v \\ -\cos u \sin v \\ \sin u \end{bmatrix} \tag{3.24}$$

Going by equations (3.22), the curvature, metric and torsion tensor forms are

$$\mathcal{K}_1 = \begin{bmatrix} \frac{1}{R} & 0 \\ 0 & \frac{1}{R} \end{bmatrix} \; \mathcal{K}_h = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \; M_1 = \begin{bmatrix} R & 0 \\ 0 & R \cos u \end{bmatrix} \; M_h = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \; T_1 = \begin{bmatrix} 0 \\ -\frac{1}{R} \tan u \end{bmatrix}^T \; T_h = \begin{bmatrix} 0 \\ 0 \end{bmatrix}^T .$$

3.2.5 Case I: Rolling contact

If the IAB rolls on the head, and for simplicity, assume that the head is a unit sphere, then the equations of contact are

$$\dot{\alpha}_1 = \begin{bmatrix} \frac{1}{2}\omega_1 \\ \frac{1}{2}\,\sec u_1\,\omega_2 \end{bmatrix}, \quad \dot{\alpha}_h = \begin{bmatrix} \frac{1}{2}\,\cos\psi\,\omega_1 - \frac{1}{2}\,\sin\psi\,\omega_2 \\ -(\frac{1}{2}\,v\sin\psi\,\sec u_h\,\omega_1 + \frac{1}{2}\,\cos\psi\,\sec u_h\,\omega_2) \end{bmatrix},$$
$$\dot{\psi} = \frac{1}{2}\,\sin\psi\,\tan u_h\,\omega_1 + \frac{1}{2}\,(\cos\psi\,\tan u_h - \tan u_1)\,\omega_2. \tag{3.26}$$

3.2.6 Case II: Sliding contact

Suppose that the head slides on the IAB, then $\{\omega_i\}_{i=1}^3 = 0$, such that we have the following contact dynamics,

$$\dot{\alpha}_1 = \begin{bmatrix} -v_1 \\ -v_2 \end{bmatrix}, \quad \dot{\alpha}_h = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \dot{\psi} = 0. \tag{3.27}$$

3.2.7 Case III: Rolling without slipping

If the relative motion between the head and the IAB consists of rolling without slipping, then we have $v_1 = v_1 = \omega_3 = 0$ so that the contact dynamics are

$$\dot{\alpha}_1 = \begin{bmatrix} -\omega_2 \\ -\omega_1 \end{bmatrix}, \quad \dot{\alpha}_h = \begin{bmatrix} -\omega_2 \\ \omega_1\,\sec u_2 \end{bmatrix}, \quad \dot{\psi} = \omega_1\,\tan u_2. \tag{3.28}$$

3.2.8 Case IV: Rotation about normal

Finally, suppose the motion is about the normal, it follows that $\omega_1 = \omega_2 = v_1 = v_2 = 0$, so that we have

$$\dot{\alpha}_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \dot{\alpha}_h = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad \dot{\psi} = \omega_3. \tag{3.29}$$

3.2.9 General Notes

If the curvature of the head is unknown, we can continually estimate it using proprioceptive sensors to measure the velocity of the IAB with respect to to its fixed reference frame and consequently with respect to the head. Tactile sensing can then be used to measure the position of the region of contact on the IAB surface. Examples for estimating the curvature of the head can be found in (Montana, 1988).

3.3 Multi-IAB Forward Kinematics

At a material point, $\mathbf{r}$, of the IAB surface in the configuration $\mathcal{B}$, the 3D position of a point based on the radial distance r from the origin and the angles ϕ and θ is given by (2.21). The configuration space of the IAB with respect to the spatial frame at a certain time can then be described by $g_{st}(\mathbf{r}) : \mathbf{r} \rightarrow g_{st}(\mathbf{r}) \in SE(3)$ while the strain state of the IAB is characterized by the strain field

$$\hat{\xi}_i(\mathbf{r}) = g_i^{-1} \frac{\partial g_i}{\partial \mathbf{r}} \in \mathfrak{se}(3) = g_i^{-1} g_i' \tag{3.30}$$

with the respective g_i's being the tangent vector at g_i such that $g_i' \in T_{g_i(\mathbf{r})}SE(3)$. For an incompressible IAB, the strain field becomes

$$g_i(\mathbf{r}) = \exp^{\|\mathbf{r}\|\hat{\xi}_i} = \mathbf{I} + \hat{\xi}_i \|\mathbf{r}\| + \frac{\hat{\omega}}{\|\omega\|^2}\left(1 - \cos(\|\mathbf{r}\|\|\omega\|)\right)\hat{\xi}_i^2 + \frac{\hat{\omega}^3}{\|\omega\|^3}\left(\|\mathbf{r}\|\|\omega\| - \sin(\|\mathbf{r}\|\|\omega\|)\right)\hat{\xi}_i^3. \tag{3.31}$$

From the derived relationship between the head contact coordinates and the relative motion (v_t, ω_t) of the IAB $i.e.$ equation (3.21), we can associate a Jacobian that maps IAB velocities to head position and orientation. This can be easily constructed since we have the form of the contact equations (3.21).

We assume that the IABs make a contact with the head throughout manipulation to ease control. In addition, we assume that the manipulation is stable and prehensile. A forward kinematic map $K_{iab_i}(\mathbf{r}_i) : \mathbb{R}^{n_i} \rightarrow SE(3)$ maps from respective IAB positions to head position and orientation. The

velocity of the head with respect to a fixed base frame in terms of IAB velocities can be written in terms of the forward kinematics Jacobian:

$$\begin{pmatrix} v_{iab_i} \\ \\ \omega_{iab_i} \end{pmatrix} = \frac{\partial K_{iab_i}}{\partial \mathbf{r}_i} \frac{d\mathbf{r}}{dt} K_{iab_i}^{-1} = \boldsymbol{J}_i(\mathbf{r}_i)\dot{\mathbf{r}}_i \tag{3.32}$$

where $\mathbf{r}_i$ is the spatial position of IAB i, and $(v_{iab_i}^T, \omega_{iab_i}^T) \in \mathbb{R}^6$ represents the linear and angular velocity of the i^{th} IAB about its screw basis. In essence, $\mathbf{r}_i \in \mathbb{R}^3$ with its rows of mapped to scalars by an appropriate choice of norm. The contact between the head and the IABs is mapped by the Jacobian

$$\boldsymbol{J}_{c_i}(\xi_h, \xi_{iab_i}) = \begin{bmatrix} \mathbf{I} & \hat{\boldsymbol{w}}(r_{c_i}) \\ \mathbf{0} & \mathbf{I} \end{bmatrix} J_{r_i}, \tag{3.33}$$

where $\boldsymbol{J}_{c_i} : \dot{\xi}_{r_i} \rightarrow \begin{bmatrix} v_{c_i}^T, & w_{c_i}^T \end{bmatrix}^T$, $r_{c_i} \in \mathbb{R}^3$ is a vector between the head reference point (e.g. the center of mass) and the contact with the i^{th} IAB, ξ_h is the position and relative orientation of the head, ξ_{iab_i} is the position and relative orientation of the i^{th} soft robot in world coordinates, $\hat{\boldsymbol{w}}(r_{c_i})$ is an anti-symmetric matrix for the vector r_{c_i} as defined in equation (2.10), and $\xi_r = (\xi_{r_1}, \xi_{r_2}, \cdots, \xi_{r_8})$ are the positions and orientations for each of the 8 IABs. where the manipulation map, G_i is made up of matrices of the form

$$G_i(\xi_h, \xi_r) = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \hat{w}(r_{c_i}) & \mathbf{I} \end{bmatrix} B_i(\xi_h, \xi_r), \tag{3.34}$$

and again r_{c_i} is a function of object orientation. The net force on the head is a sum of the individual forces arising from each IAB. Owing to the linearity of each individual IAB's contact force, the resultant head force can be stitched together to form G, $i.e.$

$$\tilde{F}_h = \begin{bmatrix} G_1, \ldots, G_8 \end{bmatrix} \begin{pmatrix} \tilde{F}_{c_1} \\ \\ \tilde{F}_{c_8} \end{pmatrix} = G\tilde{F}_c, \tag{3.35}$$

where $F_h \in \mathbb{R}^6$ and $F_c \in \mathbb{R}^{m_1} \times \mathbb{R}^{m_2} \times \ldots \times \mathbb{R}^{m_8}$. The *internal* or *null forces* is captured by the null space $\mathcal{N}(G)$ of the manipulation map G; this forces correspond to zero net force on the head of the patient. Each $\tilde{F}_{c_i}$ in (3.35) is of the form (3.15).

Following (Murray and Sastry, 1990), we define the velocity constraint dual of (3.34) as the constraint between the relative velocity of the head and that of the twist velocities of the contact point

$$\begin{pmatrix} \tilde{v}_{c_i} \\ \tilde{\omega}_{c_i} \end{pmatrix} = \begin{bmatrix} \mathbf{I} & \hat{\omega}(r_{c_i}) \\ 0 & \mathbf{I} \end{bmatrix} \begin{pmatrix} v_{c_h} \\ \omega_{c_h} \end{pmatrix}. \tag{3.36}$$

For a conjugate twist vector $(v_c^T, \omega_c^T)^T$ to the the forces exerted by the IABs, f_c, we have the following

$$\begin{pmatrix} v_c \\ \omega_c \end{pmatrix} = G^T \begin{pmatrix} v_{c_h} \\ \omega_{c_h} \end{pmatrix}. \tag{3.37}$$

Given a *selection matrix* $B_i^T(\xi_h, \xi_{iab_i}) \in \mathbb{R}_i^m$ for a particular manipulation task, where m_i is the range of all the forces and moments for the chosen contact primitive (or union of contact primitives), the *manipulation map* for the i^{th} IAB can be written as,

$$G_i^T(\xi_h, \xi_{iab_i})\xi_h = B_i^T(\xi_h, \xi_{iab_i})\boldsymbol{J}_{c_i}(\xi_h, \mathbf{r}_{r_i})\dot{\xi}_{iab_i} \tag{3.38}$$

where $\boldsymbol{J}_{c_i}$ is the contact Jacobian for the i^{th} soft robot, and ξ_h denotes the velocity of the head. In the arrangement of Figure 3.1, for the 8 soft robots, the manipulation constraint of the system can be written as

$$\begin{bmatrix} G_1^T \\ G_2^T \\ \vdots \\ G_8^T \end{bmatrix} \begin{pmatrix} v_h \\ w_h \end{pmatrix} = \begin{bmatrix} B_1^T \boldsymbol{J}_{c_1} & 0 & \cdots & 0 \\ 0 & B_2^T \boldsymbol{J}_{c_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & B_8^T \boldsymbol{J}_{c_8} \end{bmatrix} \begin{pmatrix} \dot{\mathbf{r}}_{iab_1} \\ \dot{\mathbf{r}}_{iab_2} \\ \vdots \\ \dot{\mathbf{r}}_{iab_8} \end{pmatrix}, \tag{3.39}$$

Below, we give examples of the composition of the head manipulation map under different scenarios on a treatment table. These would be helpful when we use (3.38) to determine the head velocity in world coordinates. In these examples, there is an implicit assumption that the angle of tilt of the head around the axis of normal is measurable by a gyroscope or a vision sensor or other sensors of similar facsimile. We show how to find the manipulation map of the head when the IAB kinematic chain underneath the head are passive, and only the four IABs surrounding the head are actuated (see Figure 3.1) $i.e.$ roll motion of the head. We then present finding the manipulation map of th ehead when all 8 IABs are simultaneously active $i.e.$ the pitch, roll and yaw motion of the head.

3.3.1 Case I: Planar and Roll Manipulation

Consider a planar manipulation of the head as shown in Figure 3.4. Again, we use the cone of forces to model a *point contact* (Nguyen, 1988) between the head and soft robots. In this case, the robots beneath the head are passive while the side robots are actuated. For each IAB, the orientation of the vector r_{c_i} with respect to the vertical, measured counterclockwise is ϕ_i when the head lies on the xy plane; the manipulation matrix maps IAB forces into x and y forces as well as a torque perpendicular to the xy plane. We have the manipulation map as

$$G_1 = \begin{bmatrix} \mathbf{I} & 0 \\ \hat{\omega} \begin{pmatrix} -r_{c_1} \sin\phi_1 \\ -r_{c_1} \cos\phi_1 \\ 0 \end{pmatrix} & \mathbf{I} \end{bmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \; G_2 = \begin{bmatrix} \mathbf{I} & 0 \\ \hat{\omega} \begin{pmatrix} -r_{c_2} \sin\phi_2 \\ r_{c_2} \cos\phi_2 \\ 0 \end{pmatrix} & \mathbf{I} \end{bmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}$$

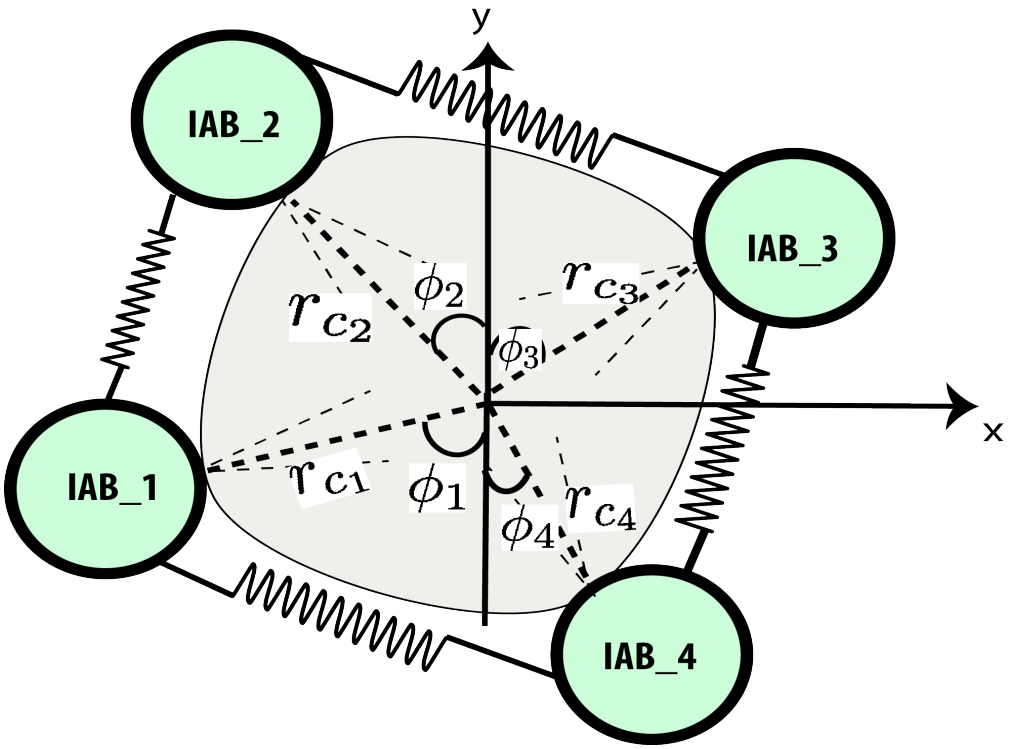


Figure 3.4: Planar head manipulation with net force on the head.

$$G_3 = \begin{bmatrix} \mathbf{I} & 0 \\ \hat{\omega} \begin{pmatrix} r_{c_3} \sin\phi_3 \\ r_{c_3} \cos\phi_3 \\ 0 \end{pmatrix} & \mathbf{I} \end{bmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix}, \; G_4 = \begin{bmatrix} \mathbf{I} & 0 \\ \hat{\omega} \begin{pmatrix} r_{c_4} \sin\phi_4 \\ -r_{c_4} \cos\phi_4 \\ 0 \end{pmatrix} & \mathbf{I} \end{bmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \tag{3.40}$$

whereupon, the planar manipulation map for the head becomes

$$G(x,y,\phi) = \begin{bmatrix} 1 & 0 & r_{c_1}\cos\phi_1 \\ 0 & 1 & -r_{c_1}\sin\phi_1 \\ 1 & 0 & -r_{c_2}\cos\phi_2 \\ 0 & 1 & -r_{c_2}\sin\phi_2 \\ 1 & 0 & -r_{c_3}\cos\phi_3 \\ 0 & 1 & r_{c_3}\sin\phi_3 \\ 1 & 0 & r_{c_4}\cos\phi_4 \\ 0 & 1 & r_{c_4}\sin\phi_4 \end{bmatrix}^T \tag{3.41}$$

for all forces with respect to the xy coordinates shown in Figure 3.4.

3.3.2 Case II: Planar Manipulation: Null Map for Zero Net Force

Now consider the case where all the vectors between each IAB in the xy plane and the head have equal magnitude such that the angle made by the head with respect to the vertical axis (y) is ϕ (see Figure 3.5). The head rolls around the vertical by an angle ϕ as shown. It follows that the manipulation map is

$$G_i(x,y,\phi) = \begin{bmatrix} \mathbf{I} & 0 \\ \hat{\omega}\begin{pmatrix} \pm r\sin\phi \\ \pm r\cos\phi \\ 0 \end{pmatrix} & \mathbf{I} \end{bmatrix} \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{pmatrix} \tag{3.42}$$

so that

$$G(x,y,\phi) = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ r\cos\phi & -r\sin\phi & -r\cos\phi & -r\sin\phi & -r\cos\phi & r\sin\phi & r\cos\phi & r\sin\phi \end{bmatrix} \tag{3.43}$$

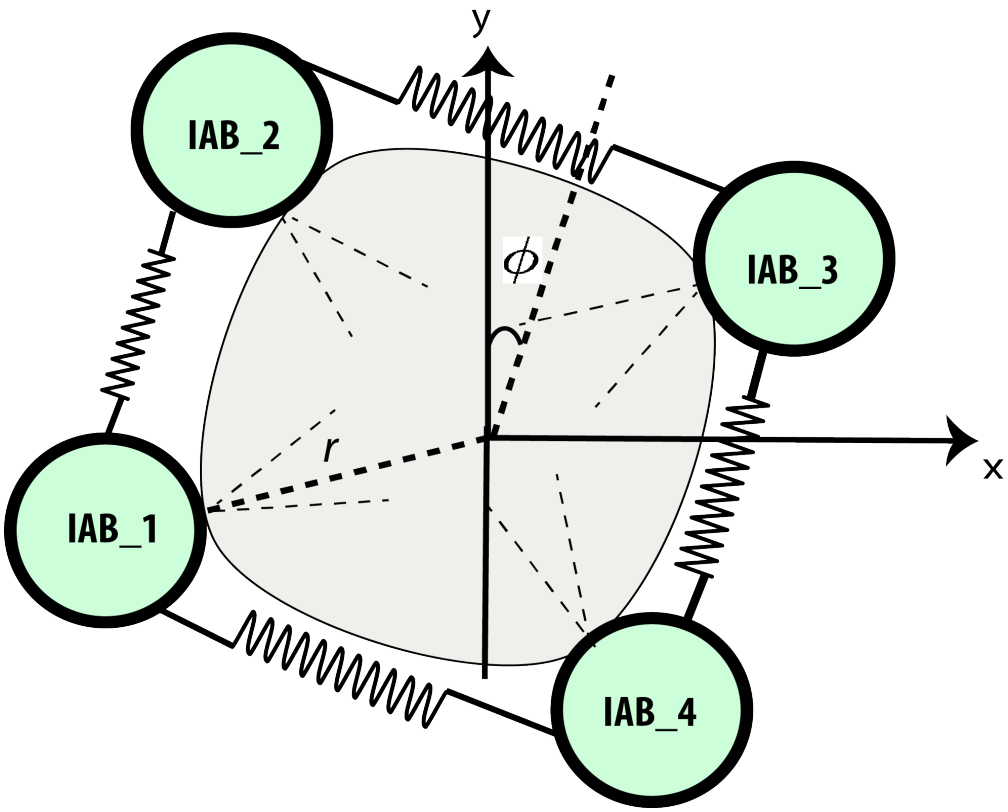


Figure 3.5: Planar head manipulation with no net force on the head.

where again all forces are measured with respect to the xy plane of Figure 3.5. The IAB's x and y forces result in an equal x and y forces on the head; the overall torque is proportional to the head orientation. When forces are applied along the line connecting diagonally oriented IABs (e.g. the lines connecting IAB_1 and IAB_3), there will be no net force on the head as the null space of the map

becomes spanned by the vector

$$\begin{pmatrix} 0 & 0 & -\frac{\sin\phi}{\cos\phi} & -r & \frac{-\sin\phi}{\cos\phi} \\ -r & 0 & -r & 0 & -r \\ 0 & -r & \frac{\sin\phi}{\cos\phi} & 0 & \frac{-\sin\phi}{\cos\phi} \\ r & 0 & 0 & 0 & 0 \\ 0 & r & 0 & 0 & 0 \\ 0 & 0 & r & 0 & 0 \\ 0 & 0 & 0 & r & 0 \\ 0 & 0 & 0 & 0 & r \end{pmatrix}. \tag{3.44}$$

3.3.3 Case III: Head Motion along xy plane and z

Consider the case where all 8 soft robots are actuated such that the head is raised to a height, h, on the treatment machine, and it is tilted along the xy plane as shown in left figure of Figure 3.6. Suppose the orientation of the head is by an angle ϕ as in Figure 3.5. We now describe the manipulation map for the IAB-head system.

$$G_i(x,y,\phi) = \begin{bmatrix} \mathbf{I} & 0 \\ \hat{\omega} \begin{pmatrix} \pm r\sin\phi \\ \pm r\cos\phi \\ h \end{pmatrix} & \mathbf{I} \end{bmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \tag{3.45}$$

so that

$$G(x,y,\phi) = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ r\cos\phi & -r\sin\phi & -r\cos\phi & -r\sin\phi & -r\cos\phi & r\sin\phi & r\cos\phi & r\sin\phi \end{bmatrix} \tag{3.46}$$

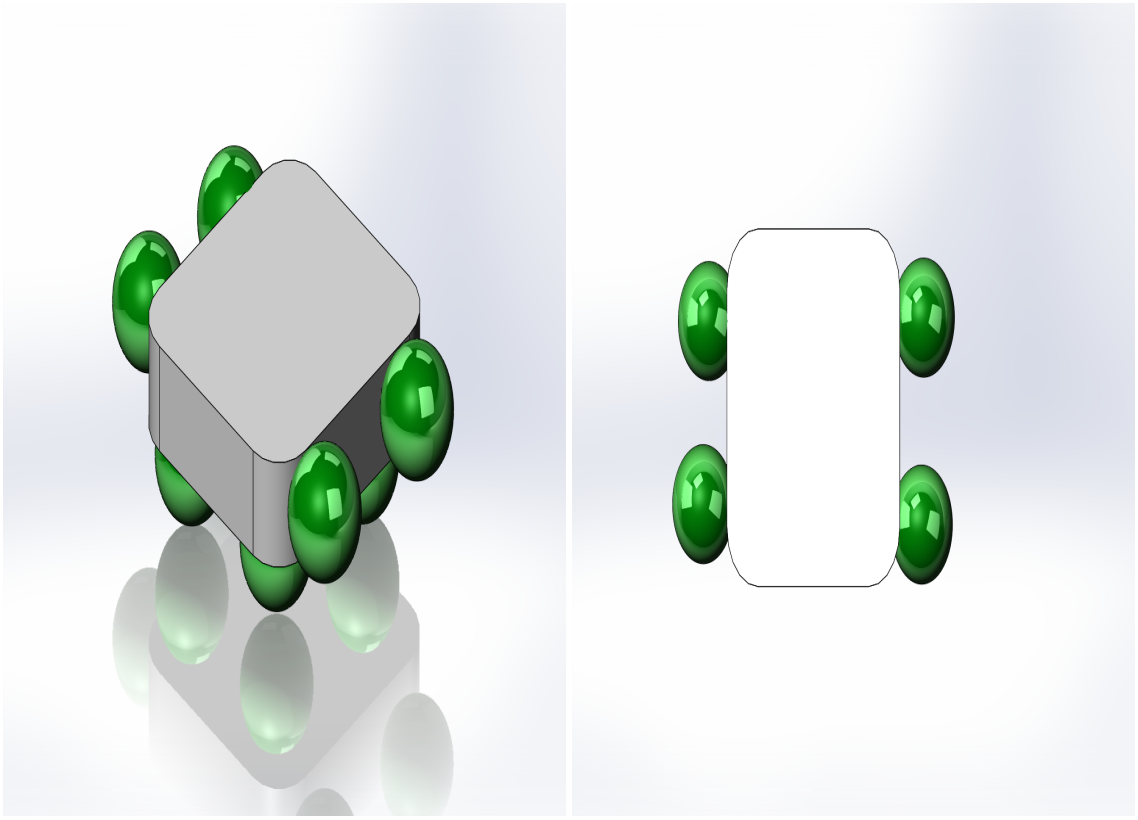


Figure 3.6: Head manipulation with all eight bladders. Head depicted with the silver-colored solid.

3.4 Multi-IAB Dynamics

From the *determinism principle for stress* (Truesdell and Noll, 1965), the Cauchy stress $\boldsymbol{\sigma}$ at any point in a material at time t for any motion up to time t determines the stress response of the material for any arbitrary motion history up to and including time t. We will derive the dynamics of the IAB system in the *strain field of the deformation*. The potential and kinetic energy of the system are considered to be derived from the constitutive strain field relations that characterize the deformation as presented in Chapter 2. Given our derivation of the strain-stress relation in equation (2.32), we now use Lagrangian deformation analysis to derive the dynamic equations of the continuum multi-IAB system of Figure 3.1.

The constitutive law which describes the macroscopic IAB material behavior with respect to a reference frame, T, at a time, t can be completely characterized by ten dependent variables viz.,

three components of the position vector, six component stress tensor variables (the shear and normal stress components), and the density, ρ, of the material (Ogden, 1997, §4.1.1).

3.4.1 Lagrange's Equations

We are interested in the final position and orientation of the IAB as a whole rather than the system of particles that characterize a deformation at every time t. For a kinetic energy T and a potential energy V, the *Lagrangian,* L, of the system in generalized coordinates is the difference between the kinetic and potential energy, $i.e.$

$$L(\mathbf{r}, \dot{\mathbf{r}}) = T(\mathbf{r}, \dot{\mathbf{r}}) - V(\mathbf{r}). \tag{3.47}$$

The equations of motion for the pneumatic system is of the form

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\mathbf{r}}_i} - \frac{\partial L}{\partial \mathbf{r}_i} = \boldsymbol{\tau}_i, \quad i = 1, \ldots, m \tag{3.48}$$

where $\boldsymbol{\tau}_i$ is the torque acting on the i^{th} generalized coordinate. Written in matrix form equation (3.48) becomes

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\mathbf{r}}} - \frac{\partial L}{\partial \mathbf{r}} = \boldsymbol{\tau}. \tag{3.49}$$

It now remains to derive the kinetic and potential energies for the IAB material. Let the velocity of an IAB material particle $\mathbf{x}$ in the current configuration at time t be $\mathbf{v}(\mathbf{r}, t)$, then the Eulerian velocity gradient tensor can be defined as

$$\boldsymbol{\Gamma} = \text{grad } \mathbf{v}(\mathbf{r}, t). \tag{3.50}$$

The first law of Cauchy's law of motion will allow us to derive the balance of mechanical energy of the system. Multiplying equation (2.36) throughout by $\mathbf{v}(\mathbf{r}, t)$, and abusing notation by dropping the arguments of $\mathbf{v}(\mathbf{r}, t)$, we find that

$$\begin{aligned} &\text{div } (\boldsymbol{\sigma}^T \cdot \mathbf{v}) + \rho \mathbf{b} \cdot \mathbf{v} = \rho \mathbf{v} \cdot \dot{\mathbf{v}} \\ \implies &\text{div } (\boldsymbol{\sigma}^T \mathbf{v}) - \mathbf{tr}(\boldsymbol{\sigma}\, \boldsymbol{\Gamma}) + \rho \mathbf{b} \cdot \mathbf{v} = \rho \mathbf{v} \cdot \dot{\mathbf{v}}. \end{aligned} \tag{3.51}$$

Following mass conservation, we integrate over volume $\mathcal{B}$ and employ the divergence theorem, so that the above relation yields the *balance of mechanical energy*:

$$\int_{\mathcal{B}} \rho \mathbf{b} \cdot \mathbf{v} \, dv + \int_{\partial \mathcal{B}} f_\rho \cdot \mathbf{v} \, da = \frac{d}{dt} \int_{\mathcal{B}} \frac{1}{2} \rho \mathbf{v} \cdot \mathbf{v} \, dv + \int_{\mathcal{B}} \mathbf{tr}(\boldsymbol{\sigma}\, \boldsymbol{\Gamma}) \, dv \tag{3.52}$$

where f_ρ is the IAB body force density, and the left hand side of the foregoing is the so-called *rate of working of the applied forces*. The symmetry of the stress tensor $\boldsymbol{\sigma}$ implies that $\mathbf{tr}(\boldsymbol{\sigma}\, \boldsymbol{\Gamma}) = \mathbf{tr}(\boldsymbol{\sigma}\, \boldsymbol{\Sigma})$ where $\boldsymbol{\Sigma}$ is given in terms of the Eulerian-strain rate tensor,Γ $i.e.$

$$\boldsymbol{\Sigma} = \frac{1}{2}(\boldsymbol{\Gamma} + \boldsymbol{\Gamma}^T) \tag{3.53}$$

so that the kinetic energy density and stress power are given by,

$$T(\mathbf{r}, \dot{\mathbf{r}}) = \frac{1}{2} \rho \mathbf{v} \cdot \mathbf{v}, \quad V(\mathbf{r}) = \mathbf{tr}(\boldsymbol{\sigma}\, \boldsymbol{\Sigma}). \tag{3.54}$$

The stress-strain relation for the IAB we have presented are only related through the deformation tensor dependence, implying that the material is Cauchy elastic. For Cauchy elastic materials, the stress power term is not conserved during deformation making integration over the material body $\mathcal{B}$ physically unrealistic (Ogden, 1997). For such materials, we may set the stored strain energy V to an arbitrary constant (e.g. an identity or $V(I) = 0$). We can derive the overall torque dynamics of an IAB system(see Appendix B) as

$$\boldsymbol{\tau} = \begin{bmatrix} \rho & 0 & 0 \\ 0 & \rho\, r^2 & 0 \\ 0 & 0 & \rho\, r^2 \sin^2 \phi \end{bmatrix} \begin{bmatrix} \ddot{r} \\ \ddot{\phi} \\ \ddot{\theta} \end{bmatrix} + \mathbf{diag} \begin{bmatrix} 2\,\rho\, r \left(\dot{\theta}\, \sin^2 \phi + \dot{\phi} \right) \\ \rho r \left(r \dot{\theta} \sin 2\phi - \dot{\phi} \right) \\ -\rho r \dot{\theta} \sin \phi \, (r \cos \phi + \sin \phi) \end{bmatrix} \begin{bmatrix} \dot{r} \\ \dot{\phi} \\ \dot{\theta} \end{bmatrix} \tag{3.55}$$

Rewriting equation (3.55) in terms of the torque for each soft robot in Figure 3.1, we have the dynamics for IAB j as

$$M_{iab_j}(r_j, \phi_j) \ddot{\mathbf{r}}_j + C_{iab_j}(r_j, \phi_j, \dot{\theta}_j, \dot{\phi}_j) \dot{\mathbf{r}}_j = \boldsymbol{\tau}_j \tag{3.56}$$

where

$$M_{iab_j} = \begin{bmatrix} \rho_j & 0 & 0 \\ 0 & \rho_j\, r_j^2 & 0 \\ 0 & 0 & \rho_j\, r_j^2 \sin^2\phi_j \end{bmatrix} \text{ and } C_{iab_j} = \textbf{diag} \begin{bmatrix} 2\,\rho_j\, r_j\, \left(\dot{\theta}_j\, \sin^2\phi_j + \dot{\phi}_j\right) \\ \rho_j r_j \left(r_j \dot{\theta}_j \sin 2\phi_j - \dot{\phi}_j\right) \\ -\rho_j r_j \dot{\theta}_j \sin\phi_j\, (r_j \cos\phi_j + \sin\phi_j) \end{bmatrix}$$

are the respective inertia and Coriolis forces matrices for the soft robot, j while $\boldsymbol{\tau}$ is the actuator torque. Since the material of the IAB is incompressible, the mass density is uniform throughout the body of the material. In general, we write equation (3.56) as

$$\boldsymbol{M}_{iab}(\tilde{\mathbf{r}})\ddot{\tilde{\mathbf{r}}} + \boldsymbol{C}_{iab}(\tilde{\mathbf{r}}, \dot{\tilde{\mathbf{r}}})\dot{\mathbf{r}} = \tilde{\boldsymbol{\tau}} \tag{3.57}$$

where $\tilde{\mathbf{r}} \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2} \times \cdots \mathbb{R}^{n_8}$ gives the generalized coordinates for all the IABs and $\tilde{\boldsymbol{\tau}}$ are the vectorized torques of the individual robots.

3.5 Newton-Euler Equations for IAB and Head System

The dynamics of the head is a form of (3.57) but without the actuator torques. In local coordinates, it has the form

$$\boldsymbol{M}_h(\zeta)\ddot{\zeta} + \boldsymbol{C}_h(\zeta, \dot{\zeta})\dot{\zeta} + \boldsymbol{N}_h(\zeta, \dot{\zeta}) = 0 \tag{3.58}$$

with ζ being a local parameterization of the position and orientation of the head, $x_h \in SE(3)$, and $\boldsymbol{N}_h$ being the gravitational and frictional forces. The head and the multi-DOF IAB system are connected via manipulation constraint $i.e.$

$$G^T(\zeta, \mathbf{r})\dot{\zeta} = \boldsymbol{J}(\zeta, \mathbf{r})\dot{\mathbf{r}}. \tag{3.59}$$

Suppose that the velocity constraint produces a virtual displacement constraint in $\delta\zeta$ and $\delta\mathbf{r}$ such that for $q = (\zeta, \mathbf{r})$, we have

$$\delta\mathbf{r} = \boldsymbol{J}^{-1}(q)G^T(q)\delta\zeta$$

the Lagrange equations become

$$\left(\frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} - (\boldsymbol{\tau}, 0)\right)\delta q = 0 \tag{3.60}$$

$$\begin{pmatrix} \frac{d}{dt}\frac{\partial L}{\partial \dot{\mathbf{r}}} - \frac{\partial L}{\partial \mathbf{r}} - \boldsymbol{\tau} \\ \frac{d}{dt}\frac{\partial L}{\partial \dot{\zeta}} - \frac{\partial L}{\partial \zeta} \end{pmatrix}^T \begin{pmatrix} \delta\mathbf{r} \\ \delta\zeta \end{pmatrix} = 0 \tag{3.61}$$

$$\left(\frac{d}{dt}\frac{\partial L}{\partial \dot{\mathbf{r}}} - \frac{\partial L}{\partial \mathbf{r}} - \boldsymbol{\tau}\right)\delta\mathbf{r} + \left(\frac{d}{dt}\frac{\partial L}{\partial \dot{\zeta}} - \frac{\partial L}{\partial \zeta}\right)\delta\zeta = 0$$

$$GJ^{-T}\left(\frac{d}{dt}\frac{\partial L}{\partial \dot{\mathbf{r}}} - \frac{\partial L}{\partial \mathbf{r}} - \boldsymbol{\tau}\right)\delta\zeta + \left(\frac{d}{dt}\frac{\partial L}{\partial \dot{\zeta}} - \frac{\partial L}{\partial \zeta}\right)\delta\zeta = 0 \tag{3.62}$$

from where

$$\left(\frac{d}{dt}\frac{\partial L}{\partial \dot{\zeta}} - \frac{\partial L}{\partial \zeta}\right)\delta\zeta + GJ^{-T}\left(\frac{d}{dt}\frac{\partial L}{\partial \dot{\mathbf{r}}} - \frac{\partial L}{\partial \mathbf{r}}\right) = GJ^{-T}\boldsymbol{\tau} \tag{3.63}$$

given the arbitrariness of $\delta\zeta$. Equations (3.63) alongside (3.59) completely describe the system.

CHAPTER 4

A SYSTEM IDENTIFICATION APPROACH TO PATIENT POSITIONING

4.1 Introduction

In this chapter, we describe our early works on the assessment of motion correction mechanisms that demonstrate proof-of-concept studies of frameless and maskless patient immobilization systems. The immobilization systems addressed here concern the identification of the complete head and IAB immobilization systems for realizing 1-DOF and 3-DOF kinematic control of a head and neck phantom. Using vision-based sensing of head and neck phantoms, we develop state-,observer-based and neuro adaptive feedback control of a generic head and neck system above a treatment table. Virtually all the work described in this chapter have previously appeared in the following IEEE publications (Ogunmolu et al., 2015b, 2016b), and (Ogunmolu et al., 2017) [1] [2] [3].

4.2 One-DOF Testbed

The goal here is to address the non-rigid motion compensation during H&N RT. We control the 1-DOF motion of th head, raising or lowering a generic patient's head, lying in a supine position, to a desired height above a treatment table. The system consists of a single inflatable air bladder (IAB), a mannequin head and a neck/torso motion simulator, two different Kinect RGB-D cameras

[1] ©2015 IEEE. Reprinted, with permission, from Olalekan Ogunmolu, Xuejun, Gu, Steve Jiang, and Nicholas Gans, A Real-Time Soft Robotic Patient Positioning System for Maskless Head-and-Neck Cancer Radiotherapy: An Initial Investigation. IEEE International Conference on Automation Science and Engineering, Gothenburg, Sweden, August 2015.

[2] ©2016 IEEE. Reprinted, with permission, from Olalekan Ogunmolu, Xuejun, Gu, Steve Jiang, and Nicholas Gans, Vision-based Control of a Soft Robot for Maskless Head and Neck Cancer Radiotherapy. In IEEE International Conference on Automation Science and Engineering, Fort Worth, Texas, August 2016.

[3] ©2017 IEEE. Reprinted, with permission, from Olalekan Ogunmolu, Adwait Kulkarni, Yonas Tadesse, Xuejun, Gu, Steve Jiang, and Nicholas Gans, Soft-neuroadapt: A 3-DOF neuro-adaptive patient pose correction system for frameless and maskless cancer radiotherapy. In IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, September 2017.

that infer the patient's position, two current-controlled pneumatic valve actuators, and a National Instruments myRIO microcontroller. Here, we extended and improve our previous work (Ogunmolu et al., 2015b) with a better vision tracking and localization of the head's position by filtering and fusing the two RGB-D estimates of the head's position and orientation. We improve on the system identification of the soft-robot system and now incorporate an optimal control system. The result is a much improved motion control.

The RGB-D sensors are mounted directly above the head for raw head position and velocity measurements, while local Kalman filters (KFs) provide two estimates of the head position and velocity. The sensor estimates are aggregated using a track-to-track KF-based sensor fusion algorithm. We apply the fusion result in a new robust control law for the pneumatic actuator valves, thereby regulating air pressure within the IAB and moving the patient's head as desired. The real-time controller was deployed on a National Instruments myRIO embedded system running LabVIEW 2015. The LabVIEW algorithms were processed within a Windows 7 virtualbox running on the Ubuntu host workstation.

In our setup, two microsoft kinect RGBD cameras measured the head displacement from the table, upon actuation, while a medical pillow around the head reduced infra-red wavelengths scattering caused by the hair on the mannequin head, and negated undesirable head rotations. We used two cameras to avoid 3D measurement errors that are prone to arise when using depth sensing. Essentially, a Kinect Xbox 360, and a Kinect for Windows v2 sensor estimated the head position and velocity (see Figure 4.1). The two sensors use different electronic perception technologies to determine distance of an object from the camera origin. They therefore have different lateral and range resolutions as well as different noise characteristics. The head pose estimation on a 32GB RAM mobile workstation with Intel Core i7-4800MQ processor running 64-bit Ubuntu Trusty on a Linux 4.04 kernel. All real-time control processing were implemented on a National Instruments myRIO embedded system running LabVIEW 2014.

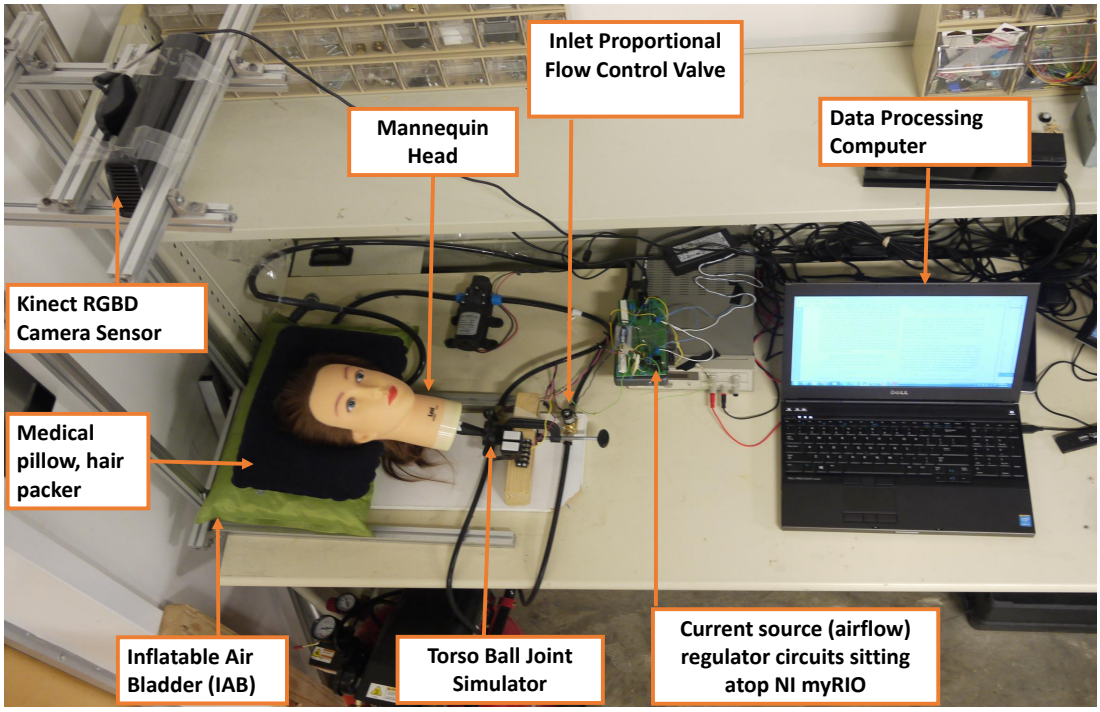


Figure 4.1: One-DOF Experimental Testbed

4.2.1 Vision-Based Head Pose Estimation

In the clinic, a very accurate stereo vision system (e.g. Align3D's VisionRT 3D surface imaging system, UK) is used to measure the patient's position on the treatment table. For our protypical verification, we adopt the Kinect Xbox and v1 cameras for laboratory testing. The v1 sensor uses the time-of-flight (ToF) perception principle, where light pulses illuminate a scene and depth is calculated by the phase shift of the returned light signals. The active infra-red reduces the dependence on ambient lighting. Both sensors acquires images at 30Hz interactive rate. The v1 has in-built noise improvement capabilities due to the limited sensor resolution that the Xbox sensor does not have (as the chart of Figure 4.2 reveals), necessitating the need for filtering the measurements from the Xbox. So we perform local Kalman Filter estimates of each sensor's observations, then fuse the estimates using a variance-weighted sensor fusion algorithm (to be described shortly).

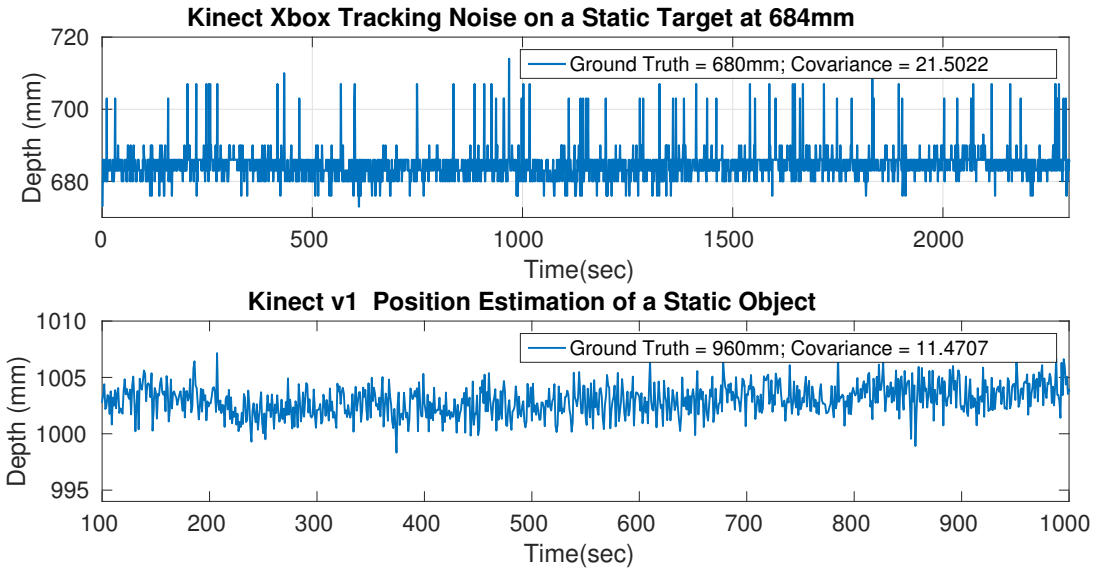


Figure 4.2: Noise Floor of Kinect Xbox Sensor vs. Kinect v1 Sensor

The face was detected using the Haar Cascade Classifiers (HCC) proposal in (Viola and Jones, 2001). To reduce memory consumption in searching for facial regions in an image, we first spatially downsampled the raw spatial image resolution from the camera. To achieve robust detection, the minimum number of neighbors in each candidate rectangle feature was determined based on our experience. The search area within an image was chosen to be within the range of (5×5) pixels and (20×20) pixels. This gave us more than 90% face detection rate for both sensors. A similar approach was used for the eye classifier. The final implementation achieved a frame rate of 15Hz for each sensor running independently on the Linux host computer.

4.2.2 Improved Pose Estimates

Our goal is to find observation estimates $\hat{\mathbf{x}}(i)$ that minimizes the mean-squared error to the true measurement $\mathbf{x}(i)$, given a measurement sequence $z(1), \cdots, z(j)$, $i.e.$

$$\begin{aligned}\hat{\mathbf{x}}(i|j) &= \arg \min_{\hat{\mathbf{x}}(i|j)\in\mathbb{R}^n} \mathbb{E}\{(\mathbf{x}(i)-\hat{\mathbf{x}})^T(\mathbf{x}(i)-\hat{\mathbf{x}})|z(1),\cdots,z(j)\} \\ &\triangleq \mathbb{E}\{\mathbf{x}(i)|z(1),\cdots,z(j)\} \triangleq \mathbb{E}\{\mathbf{x}(i)|Z^j\} \end{aligned} \tag{4.1}$$

where the obtained estimate is the expected value of the state at time i given observations up to time j. The covariance of the estimation error is given by

$$\mathbf{P}(i|j) \triangleq \mathbb{E}\{(\mathbf{x}(i) - \hat{\mathbf{x}}(i|j))^T (\mathbf{x}(i) - \hat{\mathbf{x}}(i|j)) \,|Z^j\}. \tag{4.2}$$

Assuming that the state model is common to both sensors, and the distance from the v1 sensor's principal point to the forehead of the patient is $d(k)$, we can form the state vector as $\mathbf{x}(k) = [d(k),\, \dot{d}(k)]^T \in \mathbb{R}^2$, and we define ΔT as the time between steps $k-1$ and k. The update equations are

$$\mathbf{x}(k) = \mathbf{F}(k)\mathbf{x}(k-1) + \mathbf{B}(k)\mathbf{u}_k + \mathbf{G}_k\mathbf{w}_k \tag{4.3}$$

where $\mathbf{F}(k) \in \mathbb{R}^{2\times 2}$ is the state transition matrix given by

$$\mathbf{F} = \begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix} \tag{4.4}$$

$\mathbf{u}(k) \in \mathbb{R}^2$ is the control input, $\mathbf{B}(k)$ is the control input matrix that maps inputs to system states, $\mathbf{G}(k) \in \mathbb{R}^{2\times 2}$ process noise matrix, and $\mathbf{w}(k) \in \mathbb{R}^2$ is a random variable that models the state uncertainty. In the absence of inputs $\mathbf{B}_k\mathbf{u}_k = 0$, and the model becomes

$$\mathbf{x}_k = \mathbf{F}_k\mathbf{x}_{k-1} + \mathbf{G}_k\mathbf{w}_k \tag{4.5}$$

where $\mathbf{w}_k$ is the effect of an unknown input and $\mathbf{G}_k$ applies that effect to the state vector, $\mathbf{x}_k$. The process noise is assumed unknown and is modeled as uncontrolled forces causing an acceleration a_k in the head position (a_k is thus a scalar random variable with normal distribution, zero mean and standard deviation σ_a). We model this into (4.3) by setting $\mathbf{G}_k$ to identity and set $\mathbf{w}(k) \sim \mathcal{N}(0, \mathbf{Q}(k))$ where the covariance matrix $\mathbf{Q}(k)$ is set to a random walk sequence defined by $\mathbf{W}_k = [\frac{\Delta T^2}{2}, \Delta T]^T$. Therefore, we find that

$$\mathbf{Q} = \mathbf{W}\mathbf{W}^T {\sigma_a}^2 = \begin{bmatrix} \frac{\Delta T^4}{4} & \frac{\Delta T^3}{2} \\ \frac{\Delta T^3}{2} & \Delta T^2 \end{bmatrix} {\sigma_a}^2. \tag{4.6}$$

Say the head displacement at time k as measured by the Xbox and v1 are $z_1(k)$ and $z_2(k)$, respectively, the sensors' measurements were mapped to the v1's reference frame and modeled as

$$z_s = \mathbf{H}_s(k)\mathbf{x}(k) + v_s(k) \qquad\qquad s = 1, 2 \tag{4.7}$$

where $\mathbf{H}_s(k) = \begin{bmatrix} 1, & 0 \end{bmatrix}^T$ maps the system's state space into the observed space, and $v_s(k) \in \mathbb{R}$ is a random variable that models the sensor error. We define $v_s(k)$ as a normally distributed random variable with zero mean and variance σ_{rs}^2 and assume that the sequences $v_1(k)$,$v_2(k)$, $\mathbf{w}(k)$ are independent and uncorrelated in time. At each time step, k, each local KF's *priori* and *posteriori* estimates were computed through the prediction and update phases,

Prediction Phase:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k-1} &= \mathbf{F}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{B}_k\mathbf{u}_k \\ \mathbf{P}_{k|k-1} &= \mathbf{F}_k\mathbf{P}_{k-1|k-1}{\mathbf{F}_k}^T + \mathbf{Q}_k \end{aligned} \tag{4.8}$$

where $\hat{\mathbf{x}}_{k|k-1}$ and $\mathbf{P}_{k|k-1}$ are the state prediction vector and the prediction covariance matrix respectively.

Update Phase:

$$\begin{aligned} \mathbf{K}(k) &= \mathbf{P}(k|k-1)\mathbf{H}(k)^T[\mathbf{H}(k)\mathbf{P}(k|k-1)\mathbf{H}(k)^T + \mathbf{R}(k)]^{-1} \\ \hat{\mathbf{x}}(k|k) &= \hat{\mathbf{x}}(k|k-1) + \mathbf{K}(k)(\mathbf{z}(k) - \mathbf{H}(k)\hat{\mathbf{x}}(k|k-1)) \\ \mathbf{P}(k|k) &= (\mathbf{I} - \mathbf{K}(k)\mathbf{H}(k))\mathbf{P}(k|k-1) \end{aligned} \tag{4.9}$$

where $\mathbf{K}(k)$, $\hat{\mathbf{x}}(k|k$, and $\mathbf{P}(k|k)$ are respectively the KF gain, *posteriori* state estimate and its state covariance matrix.

Figures 4.3 and 4.4 show the local filter estimate results of the observation from both the Kinect Xbox and v1 sensors post-filtering. The noise floor becomes noticeably reduced by each sensor after the KF filtering. The steady-state performance of both sensors include a reduction in the variance of the observation sequence by 80.81%, while the Kinect v1 shows an improvement in noise rejection by almost 60% .

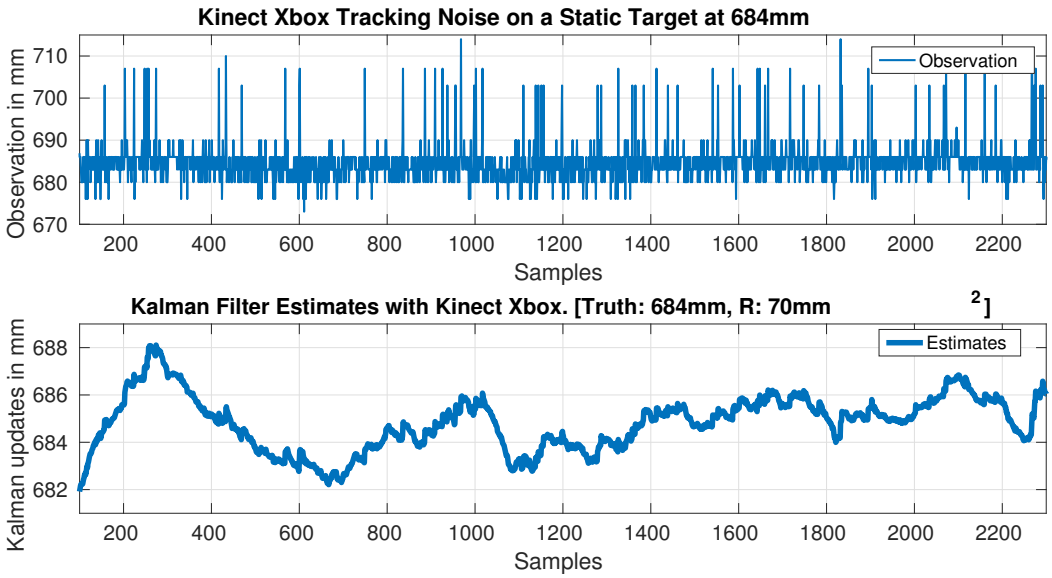


Figure 4.3: KF results for the Xbox observation

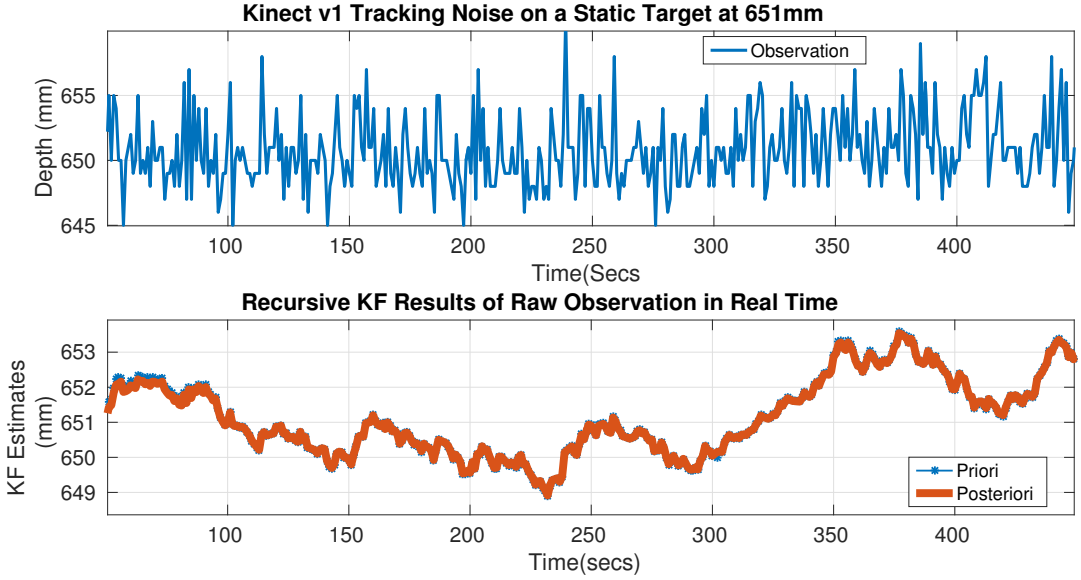


Figure 4.4: Kalman Filter on Kinect v1's Observation

4.2.3 Sensor Fusion

We combined the two filtered estimates into a *track-to-track* fused global estimate at a central fusion site. We passed each estimate through unix named pipes, and again assume a state model common

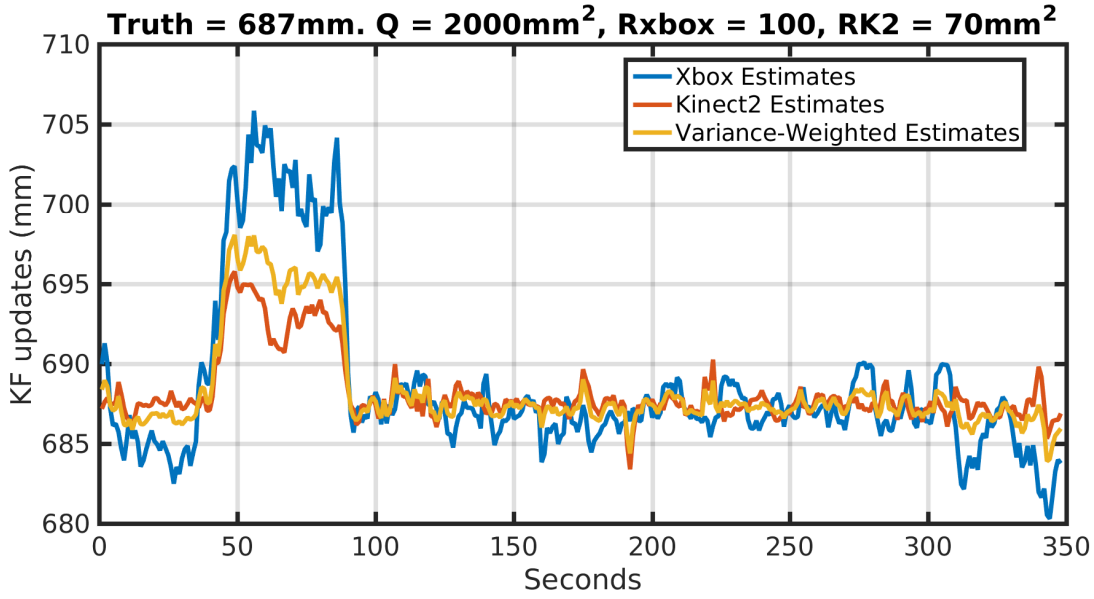


Figure 4.5: Track-to-Track Fusion of Sensors' Local Track Estimates.

to both sensors; we adopt a variance-weighted average of each local track in the global track fusion algorithm as proposed by (Durrant-Whyte, 2001)

$$\hat{\mathbf{x}}(F)(k|k) = \mathbf{P}(F)(k|k) \sum_{i=1}^{N} \left[\mathbf{P}(s)^{-1}(k|k)\hat{\mathbf{x}}(s)(k|k)\right]$$

$$\text{where } \mathbf{P}(F)(k|k) = \left[\sum_{i=1}^{N} \mathbf{P}(s)^{-1}(k|k)\right]^{-1}.$$

Figure 4.5 illustrates the fusion scheme results against single Kalman filters results during a head-raising motion. The fusion of the local tracks produces better estimates, with improved signal to noise ratio. The fused estimate assigns more weight to the less noisy signal from Kinect v1. Through the implementation of the local tracks and a global track KF estimator, we improved the accuracy of the effective signal to be used in our control algorithm to no more than a standard deviation of 0.75mm from the true position of an object. The noise spikes in the fused tracks when the process state estimates are yet to converge (as seen in Figure 4.5) can be attributed to the noisy initialization of pixels in the sensors before they attain their steady state values. On average, it takes approximately 30 seconds for the pixel values in the Kinect sensor to reach their final steady state

values (Andersen et al., 2012). This can be avoided by running the fusion algorithm for at least 2 minutes before the fused signal is used for any control purposes.

4.2.4 Parametric Identification and Control

Having found reliable sensor estimates, we proceed to identify the model of the system dynamics. We collect input-output data points, and fit a model using the prediction error model (PEM) (Ljung, 1999). The PEM model works by finding the optimal parameters, $\theta^{\star}$ through the minimization of an appropriate cost function

$$G(t) = \arg\ \min_{\theta}\ V_N(\theta, Z^N)$$

where $V_N(\theta, Z^N) = \sum_{k=1}^{\mathcal{K}} \sum_{i=1}^{n} \frac{1}{2}(\hat{y}_i(k) - y_i(k))^2$, and $Z^N = \{u(1), \cdots, u(N)\, y(1), \cdots, y(N)\}$ is the vector of past input and output (fused estimates) measurements over an interval $[1, N]$. We defined a state space realization after the least-squares minimization of the above criterion so that we obtained the following second-order approximation of the system,

$$\begin{aligned} \mathbf{x}(k + Ts) &= \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{K}\mathbf{e}(k) \\ \mathbf{y}(k) &= \mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k) + \mathbf{e}(k) \end{aligned} \tag{4.10}$$

where Ts is the sampling period, $\mathbf{e}(k)$ is the modeled zero-mean Gaussian white noise with non-zero variance,

$$\begin{aligned} \mathbf{A} = \begin{bmatrix} 0 & 1 \\ -0.9883 & 1.988 \end{bmatrix},\ \mathbf{B} = \begin{bmatrix} -3.03e-07 \\ -4.254e-07 \end{bmatrix} \\ \mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix},\ D = 0, \text{and } \mathbf{K} = \begin{bmatrix} 0.9253 & 0.9604 \end{bmatrix}^T, \end{aligned} \tag{4.11}$$

with (A, B) being stabilizable and (A, C) being detectable. We then applied the standard LQG controller on the model of (4.11) by minimizing the cost,

$$J = \sum_{k=0}^{\mathcal{K}} x^T(k)\, Q\, x(k) + u(k)^T\, R\, u(k) + 2x(k)^T\, N\, u(k), \tag{4.12}$$

where $\mathcal{K}$ is the terminal sampling instant, Q is a symmetric, positive semi-definite matrix that weights the n-states of the A matrix, N specifies a matrix of appropriate dimensions that penalizes the cross-product between the input and state vectors, while R is a symmetric, positive definite weighting matrix on the control vector u. The quadratic cost function in (4.12) allows us to find an analytical solution (controller sequence) to the minimization of J over the prediction horizon, n_y

$$\Delta u = \arg\min_{\Delta u} J \tag{4.13}$$

where Δu is the future control sequence and the first element in the sequence is used in the control law at every time instant. Choosing Q, and R in (4.12) as

$$Q = \begin{bmatrix} 1.0566 & 0 \\ 0 & 1.0566 \end{bmatrix}, \quad R = \begin{bmatrix} 0.058006 \end{bmatrix}. \tag{4.14}$$

We then construct a full online estimator for the identified plant as in Figure 4.6, whereby the noise processes are assumed to be independent, white, Gaussian, of zero mean and known covariances. The optimal controller gains, K_{opt}, are determined from

$$K_{opt} = R^{-1}(B^T P + N^T) \tag{4.15}$$

(Anderson and Moore, 1990) where P is the solution to the algebraic Riccati equation earlier and $\mathbb{E}[w(k)w'(\tau)] = R(k)\delta(k-\tau)$. Therefore, the online optimal estimate, $\hat{x}(k+1)$ of $x(k)$ is

$$\hat{x}(k+1) = A(k)\hat{x}(k) + K_{lqg}\left[C(k)\hat{x}(k) - y(k)\right] \tag{4.16}$$

where $\hat{x}(k_0) = \mathbb{E}\left[x(k_0)\right]$. The observer is equivalent to a discrete stochastic Kalman filter that estimates the optimal state $\hat{x}(k|k)$ as shown in Figure 4.6. The online, unbiased estimate is

$$\hat{x}(k+1) = A(k)\hat{x}(k) - K_{obs}[C(k)\hat{x}(k) - y(k)] + B(k)u(k). \tag{4.17}$$

Figure 4.7 shows the results. We notice a settling time of approximately 24 seconds before we reach steady state. The delay arises from our design requirements and is not a drawback in clinical

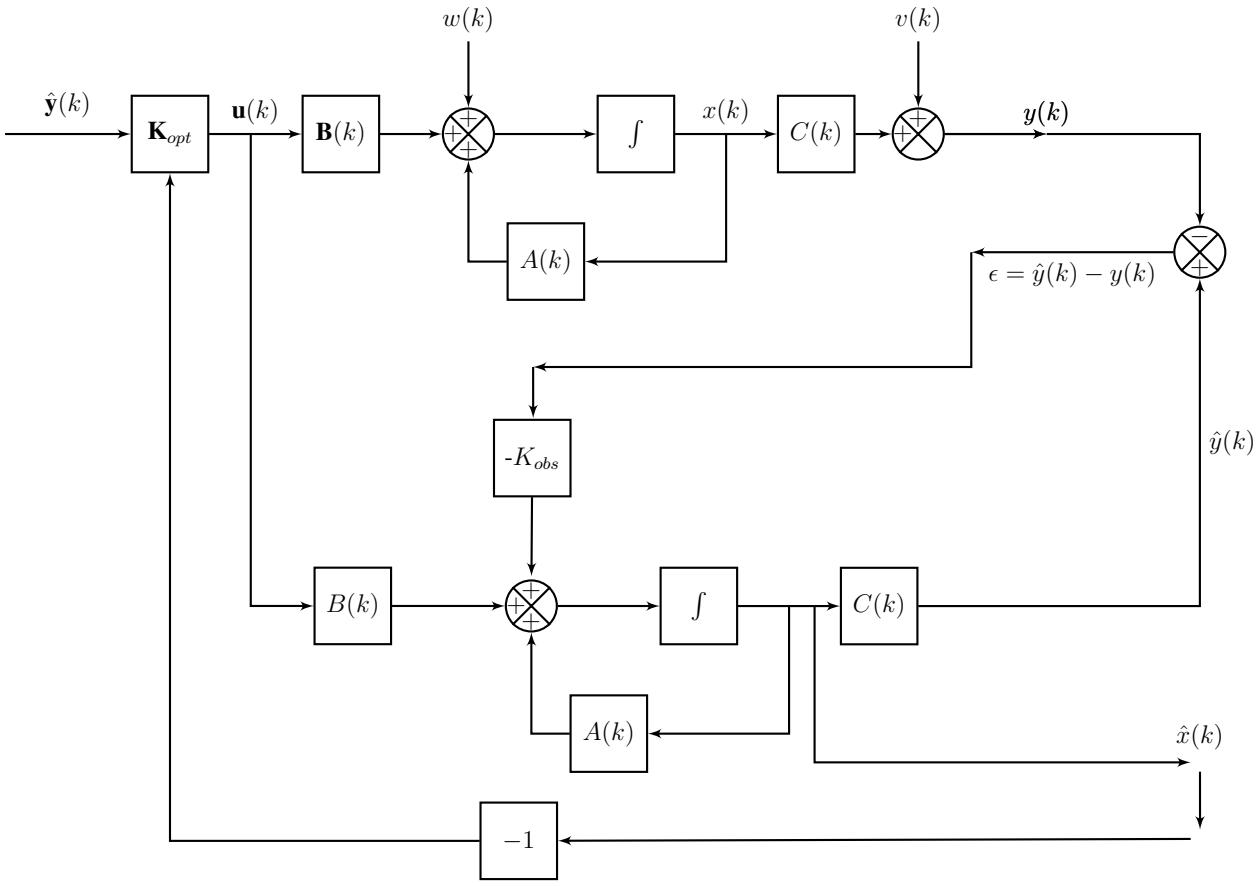


Figure 4.6: Full Linear Quadratic Gaussian Plant Estimator

trajectory tracking where we must ensure smooth head motion to desired target. It is also seen that the controller exhibits relatively smooth tracking within a 1.5 mm standard deviation over time after a relative overshoot of 5mm in bottom graph of Figure 4.7. The overshoot can be explained by the estimator's search for a steady state region based on the time it takes for the pixel values of the sensors to reach steady state. The controller tracks the reference to within $\pm 2mm$.

4.3 Three DOF System

Here, we use 3 collocated soft robots around a custom-designed mannequin head to test the feasibility of the head immobilization system. We propose a model-based and a supervised learning approach. We present **Soft-NeuroAdapt**, a set of three **soft** actuators that employ a **neuro**-controller

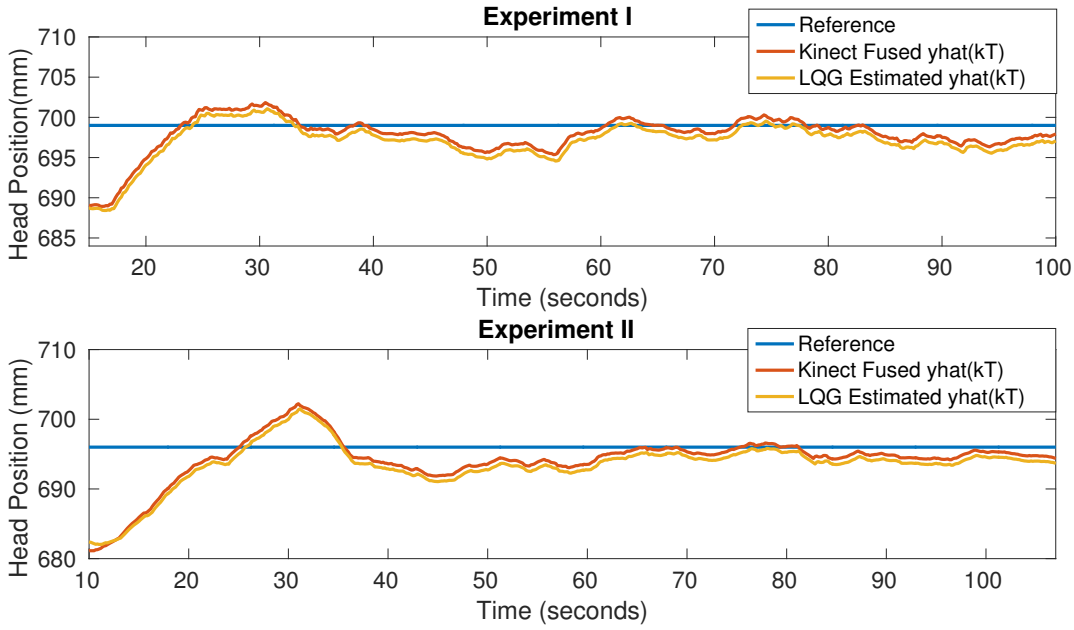


Figure 4.7: LQG Controller Position Compensation Results.

to **adapt**ively compensate for positioning deviations. Exhibiting highly nonlinear dynamics, controlling soft robots for precise actuation is complicated. While a closed-form model may work in highly-structured environments, system parameters and dynamics change with different patients' head and upper torso anatomy, creating the need for re-estimating system parameters. Our goal is to derive a learning-based controller, going beyond task-specific, expert-driven methods in order to adaptively generalize to new control systems. For such higher-level tasks, it suffices to learn the underlying system dynamics in real-time, than building hand-engineered, finite-state machines to implement intended behaviors.

Following our prior investigative studies, (Ogunmolu et al., 2015a, 2016a), on 1-DoF soft-robot compensation systems, we present a 3-DoF soft robot system that addresses 3-DoF involuntary intrafractional motions of the head and neck (H&N) region during F&M RT. In what follows, we present a neural network dynamics estimator that learns a system model, then adapts to the model in its control law to provide bounded tracking of set trajectory. The controller corrects intrafractional (involuntary) patient motions along three defined axes namely head pitch, roll and elevation angles

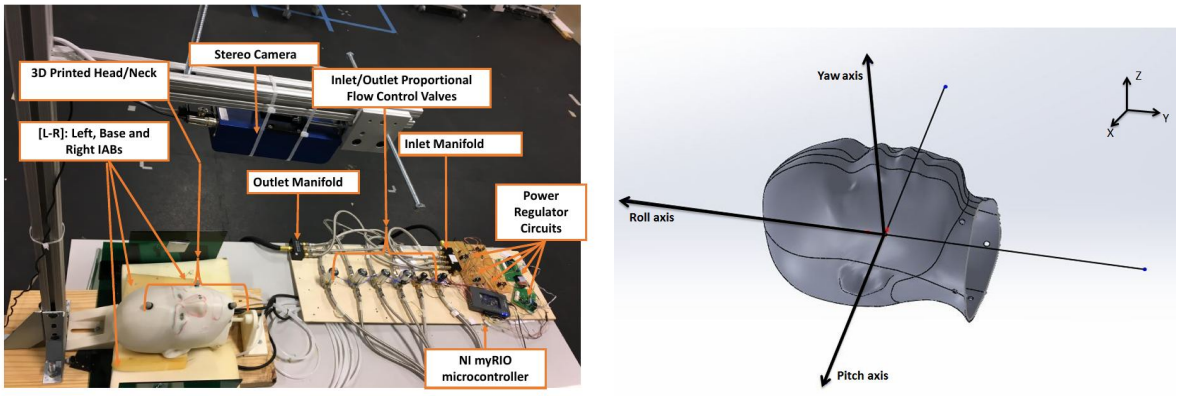


Figure 4.8: Hardware Setup and Head Coordinate System

for a patient lying in a supine position on a table. We use three inflatable air bladders (IABs), actuated through a system of inlet and outlet solenoid proportional valves. The controller uses state feedback to provide bounded stability of states, a reference trajectory component to provide command tracking and a neural-network component to adaptively converge states that start outside of the sphere of stability into the region of stability. We perform head pose tracking in the 3D space of a stereo-camera, and we conduct experiments to validate the proposed bio-pneumatic system and controller.

4.3.1 Hardware Overview

The actuation mechanism consists of three custom-designed elastomeric IABs. The base IAB (see Figure 4.8) is 180mmx280mm when flat and inflates to a maximum height of $\sim$ 75 mm, while the other two are 180mmx140mm in size. The IABs consist of inflatable rubber, encased in a breathable foam pad for comfort, modified to be the size of an average adult male head. The IABs have separate inlet and outlets openings, connected with crack-resistant polyethylene tubing (1/8" ID and 1/4" OD), which sustains pressure of up to 32psi.

Each hose leads to a proportional solenoid valve, which is connected to rectangular manifolds (one manifold to the inlet supply and the second to the outlet supply). We use six Dakota Instruments EM valves (Model PSV0105, Orangeburg, NY, USA) to supply proportional torques to the soft

actuators. A regulated air canister supplied constant air pressure at 15 psi to the inlet-air conveying manifold, while a suction pump supplied vacuum pressure at 12 psi to the valves that removed air from the bladders. The air rate of flow into or out of each bladder was controlled via custom-built voltage regulating circuits which got PWM signals from a National Instruments (NI) myRIO microcontroller.

We 3D printed a custom manikin head, measuring $155 \times 240 \times 200$ mm ($W \times L \times D$), and comparing between 50% and 75% weight of a typical adult male head or 99% of a typical adult female head. The head was fitted with a ball-joint in the neck to replicate motion of the human head about the neck. An Ensenso 3D camera is mounted approximately 45° above the head to measure the pose of the head in real time. All vision processing, systems modeling and control laws were computed on a CORSAIR PC. We exchange the neuro-control and sensor signals via the publish-subscribe IPC of the ROS middleware installed on the PC. Adaptive control inputs were sent via udp packets to the RIO microcontroller. The hardware is shown on the left inset of Figure 4.8. The reference frame of the head (right inset of Figure 4.8)is described as follows: the pitch/x-axes points from the left ear out of the right ear, the yaw/z-axes points from the back of the head through the forehead through, and the roll/y-axes goes from the neck through the top of the head. The left and right bladders control the roll angles/x-axes motions while the bladder underneath the head, henceforth referred to as the base bladder, controls the pitch angles and z-axis motions.

4.3.2 Vision-based Pose Estimation

We acquired a 3D camera from Ensenso GmbH (model N35) to reconstruct the surface image and measure head pose. The N35 camera captures multiple image pairs during exposure; each pair is made up of different patterns – controlled by piezo-actuators. A stereo-matching algorithm then gathers information from all image pairs after capture to produce a high-resolution point cloud of the scene (Ensenso, Ensenso). We mounted the 3D sensor such that its lens faced the head at

approximately 45° from the vertical during experiments. We seek to control the motion of the head about three axes, namely z, pitch and roll axes.

The dense point cloud of the scene has (i) marked jump in rendered points along the z-axis of the camera due to the single view angle by the camera; (ii) the scene clutter and lack of multiple camera view angles does not affect the representation of the face; (iii) thus, through spatial decomposition of the scene, we can separate the face from the scene. However, the point cloud is computed from monochromatic IR image pairs (with no texture information) making morphological operations difficult. Due to the multiple image pairs used in 3D reconstruction to generate a highly accurate measurement, the camera is limited to a maximum frame rate of 10Hz. Inspired by Rusu's work (Rusu, 2009), we divide the segmentation problem into stages, with each stage involving segmenting out candidates that do not belong to the object we want to identify (the frontal face) in the scene. Our engineering philosophy in the segmentation phase is inspired by spatial decomposition methods that determine subdivisions and boundaries to allow retrieval of data that we want given a measure of proximity. In this case, we know that the location of the table cannot exceed a given height during experiments, and the camera's position is fixed while the head moves based on bladders' actuation. Separating objects that represent planar 2D geometric shapes from the scene therefore simplifies the face segmentation algorithm. By finding and removing objects that fit primitive geometric shapes from the scene, clustering of the remaining objects would yield the face of the patient in the scene. We fit a simplified 2D planar object to the scene such that searching for points $\mathbf{p}_i \in \mathcal{P}$ that support a 2D plane can be found within a tolerance defined by the inequality $0 \leq |d| \leq |d_{max}|$, where $|d_{max}|$ represents a user-defined threshold to segment out (Rusu, 2009).

We proceed as follows: (i) the point cloud of the scene was acquired from the computed disparity map of the two raw camera images; (ii) to minimize sensor noise whilst preserving 3D representation, the acquired point cloud was downsampled using a SAmple Consensus (SAC)-based robust moving least squares algorithm (RMLS) (Rusu, 2009); (iii) we then searched for the edges of 2D planar regions in the scene with Maximum Likelihood SAmple Consensus (MLESAC)(Torr,

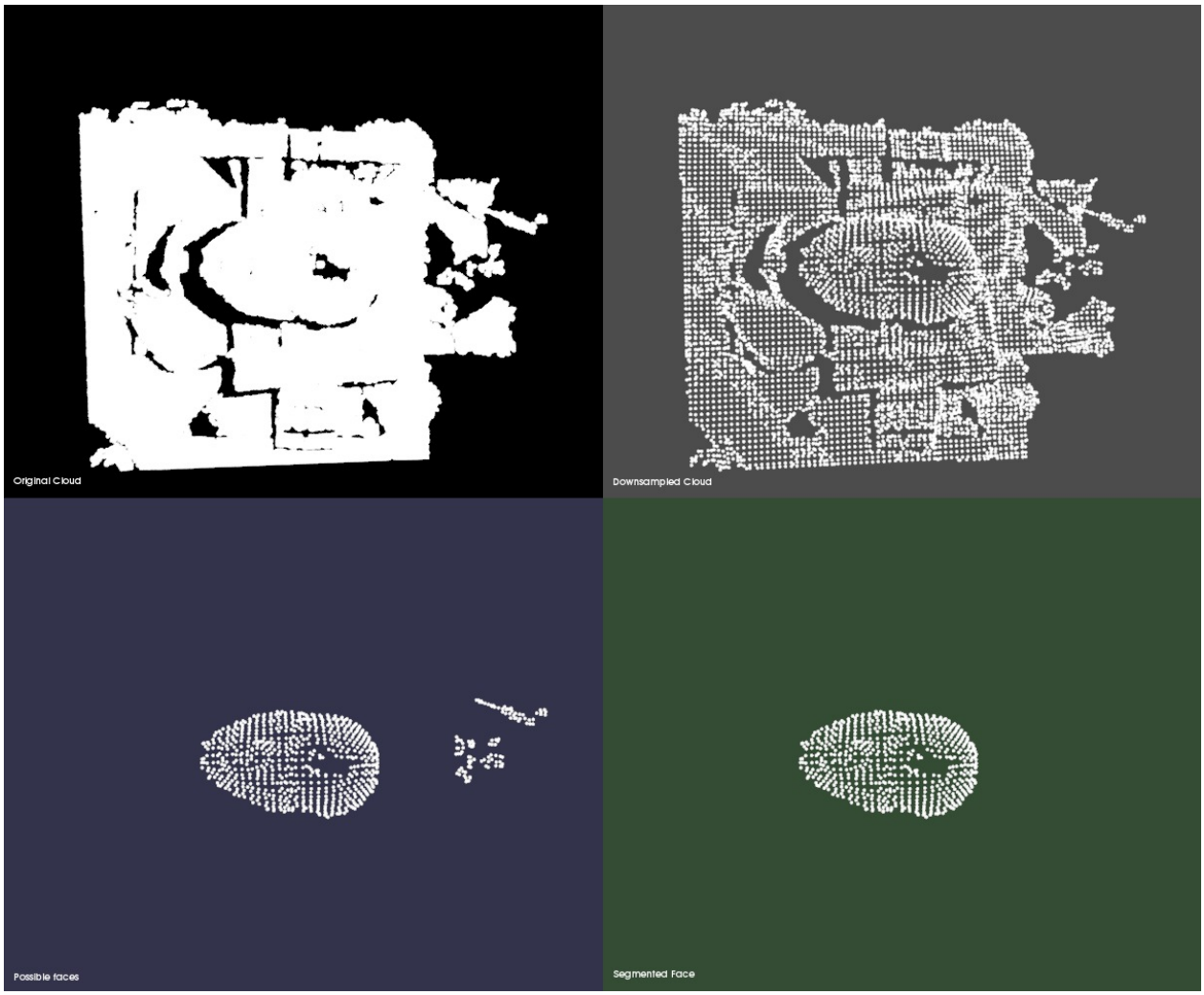


Figure 4.9: Point Cloud Segmentation

2000), and we bound the resulting plane indices by computing their 2D convex hull; (iv) model fitting stage extrudes the computed hull (of objects lying above the 2D planar region) into a prism model based on a defined L_1 Manhattan distance; this gives the points whose height threshold is about the region of the face in the scene (Rusu et al., 2008); (v) we then cluster the remaining points based on a heuristically determined L_2 distance between points remaining within the polygonal plane. The largest cluster gives us the face.

The segmentation result is presented in Figure 4.9. A dense point cloud of the experimental setup scene is shown in the top-left cornet of the image. This cluttered cloud was downsampled to

the top right cloud of the figure. Using RANSAC, we searched for 2D plane candidates in the scene and compute the convex hull of found planar regions; the ensuing results is shown in bottom left inset of the figure. We then extrude point indices within the hull into a prismatic polygonal model to obtain the face region (shown in the bottom-right of Figure 4.9. An additional step clusters the resultant cloud based on a Euclidean distance. The largest cluster is taken to be the face.

We now describe these segmentation stages. The downsampling algorithm is an implementation of (Rusu, 2009, §6). We first normalize the coordinates of the original point cloud, $\mathcal{P}$, ensuring that the distance between points $p \in \mathcal{P}$ is upper-bounded by 1 based on the diagonal of $\mathcal{P}$'s bounding box. It computes a weighting factor, α, given by

$$\alpha = \mu_x + k \cdot \sigma_x, \tag{4.18}$$

where μ and σ respectively denote the mean and standard deviation of the mean distance distribution between points and k is a user-chosen variable. Through SAC, an estimate $\hat{\mathcal{P}}$ of the original point cloud is computed and represented as a set of equidistant grid points in the neighborhood of $\mathcal{P}$. The points of $\mathcal{P}$ are then projected to a local plane of reference through their k nearest neighbors to assure proximity to the surface of $\mathcal{P}$. Points $\hat{p}_i \in \hat{\mathcal{P}}$ are fitted to the surface that approximates $\mathcal{P}$ with a bivariate polynomial height function in a local Darboux frame (with orthonormal axes **u**, **v**, **n**; **v** is chosen to be parallel to the local reference frame's normal). The polynomial weights are computed for the k nearest neighbors of **q** as

$$w_i = \exp\left(-\frac{\|\hat{\mathbf{p}} - \mathbf{p}_i\|^2}{\alpha}\right). \tag{4.19}$$

Surfaces in the proximity of $\hat{p}$'s neighbors are approximated using

$$n_{(u,v)} = \sum_{i=1}^{N} c_i . f^i_{(u,v)}, \tag{4.20}$$

where $u, v,$ and n are coordinates along the Darboux frame axes, $f^i_{(u,v)}$ are height function members of bivariate polynomials. The result of the resampling algorithm is shown in the top-right image of Figure 4.9. Further details of the segmentation algorithm can be found in (Ogunmolu et al., 2017).

4.3.3 Head Pose Estimation

With the facial point cloud segmented, we defined three points on the head. Our goal is to compute the optimal translation and rotation of the head from a model point set $\mathbf{X} = \{\overrightarrow{x}_i\}$ to a measured point set $\mathbf{P} = \{\overrightarrow{p}_i\}$, where $N_x = N_p = 3$, and the point $\overrightarrow{x}_i \in \mathbf{X}$ has the same index as $\overrightarrow{p}_i \in \mathbf{P}$. All point coordinates are with respect to a Cartesian frame in the stereo camera. We consider the world frame to have an origin at the centroid of the clustered point cloud of the head when all IABs are at ambient pressure, with the axes oriented as described in figure to the right of this page. Following the approach of (Besl, Paul J.; McKay, 1992), we compute the cross-covariance matrix of $\mathbf{P}$ and $\mathbf{X}$ as Σ_{px}, extract the cyclic components of this skew symmetric matrix as $\mathbf{\Delta}$, and use it to form the symmetric 4×4 matrix $\mathbf{Q}(\mathbf{\Sigma}_{px})$ as follows,

$$\mathbf{Q}(\mathbf{\Sigma}_{px}) = \begin{bmatrix} \mathbf{tr}(\mathbf{\Sigma}_{px}) & \mathbf{\Delta}^T \\ \mathbf{\Delta} & \mathbf{\Sigma}_{px} + \mathbf{\Sigma}_{px}^T - \mathbf{tr}(\mathbf{\Sigma}_{px})\mathbf{I}_3 \end{bmatrix}. \tag{4.21}$$

The unit eigenvector, q_R, that corresponds to the maximum eigenvalue of $\mathbf{Q}(\mathbf{\Sigma}_{px})$ is selected as the optimal rotation quaternion; we find the optimal translation vector as

$$\overrightarrow{q}_T = \overrightarrow{\mu}_x - \mathbf{R}(\overrightarrow{q}_R)\overrightarrow{\mu}_p \tag{4.22}$$

where μ_x and μ_p are the mean of point sets $\mathbf{X}$ and $\mathbf{P}$ respectively. Obtaining the roll, pitch and yaw angles from q_R is trivial and the pose of the face is described by tuples $[q_T, q_R] = \{x, y, z, \theta, \phi, \psi\}$ with respect to the world frame. Given the 3-DOF setup, we choose to control three states of the head: z, θ, ϕ ($i.e.$ z, roll, and pitch).

4.3.4 Adaptive Neuro-Control Formulation

Our primary theoretical contribution in this chapter is the approximation of the nonlinear system by a long short-term memory (LSTM) (Hochreiter and Schmidhuber, 1997), equipped with an adequate number of neurons in its hidden layers. We parameterized the last layer of the network

with a fully connected layer that outputs control torques to the valves. The neural network can be seen as a memory-based model that remembers effective controls for the adaptation mechanism in the presence of uncertainties and external disturbance.

Following our previous approach in (Ogunmolu et al., 2016a, §IV.B), we fix a persistently exciting input signal $u_{ex} \in L_2 \cap L_\infty$ that excites the nonlinear modes of the system. We then parameterized the system with a neural network with sufficient number of neurons. The neural network (NN) provided information on the changing parameters of the system during control trials. The adjustment mechanism is computed from inverse Lyapunov analysis, where we choose adaptive laws that guarantee a nonpositive-definite Lyapunov function derivative when evaluated along the trajectories of the error dynamics.

The neural network parameterizes the nonlinear dynamical system $f(\cdot)$, and maps the parameterized model to appropriate valve torques. We add a feedforward and feedback term in the overall controller (to be introduced shortly) that guarantee system stability and robustness to uncertainties. Therefore, the global controller keeps the states of the system bounded under closed-loop dynamics, ensuring convergence to desired trajectories from states that are initialized outside the domain of attraction, and guarantees robust reference tracking in the presence of non-parametric uncertainties.

For the multi-input, multi-output (MIMO) adjustable system,

$$\dot{\mathbf{y}} = \mathbf{A}\mathbf{y} + \mathbf{B}\mathbf{\Lambda}\left(\mathbf{u} - f(\mathbf{y}, \mathbf{u})\right) + \mathbf{w}(k) \tag{4.23}$$

where $\mathbf{y} \in R^n, \mathbf{u} \in R^m$ are known input and output vectors, and $\mathbf{A} \in R^{n\times n}, \mathbf{\Lambda} \in R^{m\times m}$ are unknown matrices, $\mathbf{B} \in R^{n\times m}, sgn(\mathbf{\Lambda})$ are known matrices, and $\mathbf{w}(k) \in R^n$ is a bounded time-varying unknown disturbance, upper-bounded by a fixed positive scalar $\mathbf{w}_{max}$, we introduce the following assumptions:

- a dynamic recurrent neural network (RNN) with N neurons, $\varphi(\mathbf{y})$, exists that maps from a compact input space $\mathbf{u} \subset \mathbb{U}$ to an output space $\mathbf{x} \subset \mathbb{Y}$ on the Lebesgue integrable functions with closed interval $[0, T]$ or open-ended interval $[0, \infty)$;

- the nonlinear function $f(\mathbf{y}, \mathbf{u})$ is exactly $\Theta^T \Phi(\mathbf{x})$ with matrix of vectorized coefficients, $\boldsymbol{\Theta} \in R^{N \times m}$, and a Lipschitz-continuous vector of basis functions $\boldsymbol{\Phi}(\mathbf{y}) \in R^N$;

- inside a ball $\mathbf{B}_R$ of known, finite radius R, the ideal NN approximation $f(\mathbf{y}) : R^n \rightarrow R^m$, is realized to a sufficient degree of accuracy, $\boldsymbol{\varepsilon}_f > 0$;

- the process noise $\mathbf{w}(k)$ is estimated alongside model parameters by the dynamic RNN;

- outside $\mathbf{B}_R$, the NN approximation error is upper-bounded by a known scalar function $\boldsymbol{\varepsilon}_{max}$ such that $\|\varepsilon\| \leq \varepsilon_{max}, \quad \forall \quad \mathbf{y} \in \mathbf{B}_R$;

- there exists an exponentially stable reference model

$$\dot{\mathbf{y}}_m = \mathbf{A}_m y_m + \mathbf{B}_m \mathbf{r}, \tag{4.24}$$

with a Hurwitz matrix $\mathbf{A}_m \in R^{n \times n}$ and $\mathbf{B}_m \in R^{n \times m}$ commanded by a reference signal $\mathbf{r} \in R^m$. For this system, we note that $n = 3$ and $m = 6$. Our objective is to design an model-reference adaptive controller (MRAC) capable of operating in the presence of parametric ($\boldsymbol{\varepsilon}_f$), and non-parametric ($\mathbf{w}(k)$) uncertainties so as to assure the boundedness of all signals within the closed-loop system. We propose the following controller

$$\mathbf{u} = \hat{\mathbf{K}}_y^{\mathbf{T}} \mathbf{y} + \hat{\mathbf{K}}_r^{\mathbf{T}} \mathbf{r} + \hat{f}(\mathbf{y}, \mathbf{u}), \tag{4.25}$$

where $\hat{\mathbf{K}}_y$ and $\hat{\mathbf{K}}_r$ are adaptive gains to be designed shortly. The $\hat{\mathbf{K}}_y^{\mathbf{T}} \mathbf{y}$ term keeps the states of the approximation set $\mathbf{x} \in \mathbf{B}_R$ stable, while the $\mathbf{K}_r^{\mathbf{T}} \mathbf{r}$ term causes the states to follow a given reference trajectory. The function approximator $\hat{f}(\cdot)$ ensures states that start outside the approximation set $\mathbf{y} \in \mathbf{B}_R$ converge to $\mathbf{B}_R$ in finite time (it converges non-parametric errors $\boldsymbol{\varepsilon}_f$ that puts certain states out of the approximation set into $\mathbf{B}_R$). We can generally write the NN model as

$$\hat{f}(y) = \hat{\boldsymbol{\Theta}}^T \boldsymbol{\Phi}(\mathbf{x}) + \boldsymbol{\varepsilon}_f,$$

where $\hat{\mathbf{\Theta}}^T$ denotes the vectorized weights of the neural network and $\mathbf{\Phi}(\mathbf{x})$ denotes the vector of inputs and outputs defined as

$$\mathbf{\Phi}(\mathbf{x}) = \{\mathbf{x}(k-d)\cdots\mathbf{x}(k-d-4)\mathbf{u}(k-d)\cdots\mathbf{u}(k-d-5)\}, \tag{4.26}$$

and ε_f is the approximation error. The closed-loop dynamics therefore become

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{\Lambda}\left(\hat{\mathbf{K}}_y^T\mathbf{y} + \hat{\mathbf{K}}_r^T\mathbf{r} + \hat{f}(\mathbf{x},\mathbf{u}) - f(\mathbf{x},\mathbf{u})\right). \tag{4.27}$$

We assume nonlinear function and approximator matching conditions, $f(\mathbf{x},\mathbf{u}) = \hat{f}(\mathbf{x},\mathbf{u})$, such that after rearrangement, (4.27) can be written as

$$\dot{\mathbf{y}} = A\mathbf{y} + B\mathbf{\Lambda}\left(\hat{\mathbf{K}}_y^T\mathbf{y} + \hat{\mathbf{K}}_r^T\mathbf{r} - \varepsilon_f(\mathbf{y})\right), \tag{4.28}$$

$$= (\mathbf{A} + \mathbf{B}\mathbf{\Lambda}\hat{\mathbf{K}}_y^T)\mathbf{y} + \mathbf{B}\mathbf{\Lambda}(\hat{\mathbf{K}}_r^T\mathbf{r} + \varepsilon_f). \tag{4.29}$$

Furthermore, we assume model matching conditions with ideal constant gains $\mathbf{K}_y$ and $\mathbf{K}_r$ so that

$$\mathbf{A} + \mathbf{B}\mathbf{\Lambda}\mathbf{K}_y^T = \mathbf{A}_m, \quad \text{and} \quad \mathbf{B}\mathbf{\Lambda}\mathbf{K}_r^T = \mathbf{B}_m, \tag{4.30}$$

from which

$$\mathbf{A} + \mathbf{B}\mathbf{\Lambda}\hat{\mathbf{K}}_y^T - \mathbf{A}_m = \mathbf{B}\mathbf{\Lambda}\tilde{\mathbf{K}}_y^T \text{ and } \mathbf{B}\mathbf{\Lambda}\hat{\mathbf{K}}_r^T - \mathbf{B}_m = \mathbf{B}\mathbf{\Lambda}\tilde{\mathbf{K}}_r^T, \tag{4.31}$$

where $\tilde{\mathbf{K}}_y^T = \mathbf{K}_y^T - \hat{\mathbf{K}}_y^T$ and $\tilde{\mathbf{K}}_r^T = \mathbf{K}_r^T - \hat{\mathbf{K}}_r^T$. The generalized error state vector $\mathbf{e}(k) = \mathbf{y}(k) - \mathbf{y}_m(k)$ has dynamics $\dot{\mathbf{e}}(k) = \dot{\mathbf{y}}(k) - \dot{\mathbf{y}}_m(k)$, so that by substituting (4.24) and (4.27) into $\dot{\mathbf{e}}$, we have

$$\dot{\mathbf{e}}(k) = \mathbf{A}_m\mathbf{e}(k) + \mathbf{B}\mathbf{\Lambda}[\tilde{\mathbf{K}}_r^T\mathbf{r} + \tilde{\mathbf{K}}_y^T\mathbf{y} - \varepsilon_f]. \tag{4.32}$$

The estimation error will be bounded as long as $\mathbf{y} \in \mathbf{B}_R$. Our goal is to keep $\mathbf{y} \in \mathbf{B}_R$.

Theorem: Given correct choice of adaptive gains $\hat{\mathbf{K}}_y$ and $\hat{\mathbf{K}}_r$, the error vector $\mathbf{e}(k)$, with closed loop time derivative given by (4.32) will be uniformly ultimately bounded, and the state $\mathbf{x}$ will converge to a neighborhood of $\mathbf{r}$.

Proof: We choose a Lyapunov function candidate $\mathbf{V}$ in terms of the generalized error state space $\mathbf{e}$, gains, $\tilde{\mathbf{K}}_y^T$, $\tilde{\mathbf{K}}_r^T$, and parameter error $\boldsymbol{\varepsilon}_f(\mathbf{y}(k))$ space ((Parks, 1966), (Landau, 1979), (Lavretsky and Wise, 2005)) as follows

$$\begin{aligned}\mathbf{V}(\mathbf{e}, \tilde{\mathbf{K}}_y, \tilde{\mathbf{K}}_r) = \mathbf{e}^T\mathbf{P}\mathbf{e} + \mathbf{tr}(\tilde{\mathbf{K}}_y^T \boldsymbol{\Gamma}_{\boldsymbol{y}}^{-1} \tilde{\mathbf{K}}_y |\boldsymbol{\Lambda}|) \\ + \mathbf{tr}(\tilde{\mathbf{K}}_r^T \Gamma_r^{-1} \tilde{\mathbf{K}}_r^T |\boldsymbol{\Lambda}|)\end{aligned} \tag{4.33}$$

where $\boldsymbol{\Gamma}_{\boldsymbol{y}}$ and $\boldsymbol{\Gamma}_r$ are fixed symmetric, positive definite (SPD) matrices of adaptation rates, $\mathbf{tr}(\mathbf{A})$ denote the trace of matrix $\mathbf{A}$ and $\mathbf{P}$ is a unique SPD matrix solution of the algebraic Lyapunov function

$$\mathbf{P}\mathbf{A}_m + \mathbf{A}_m^T\mathbf{P} = -\mathbf{Q}, \tag{4.34}$$

where $\mathbf{Q}$ is a SPD matrix. Take the time derivative of (4.33)

$$\begin{aligned}\dot{\mathbf{V}}(\mathbf{e}, \tilde{\mathbf{K}}_y, \tilde{\mathbf{K}}_r) = \dot{\mathbf{e}}^T\mathbf{P}\mathbf{e} + \mathbf{e}^T\mathbf{P}\dot{\mathbf{e}} + 2\mathbf{tr}(\tilde{\mathbf{K}}_y^T \boldsymbol{\Gamma}_{\boldsymbol{y}}^{-1} \dot{\tilde{\mathbf{K}}}_y |\boldsymbol{\Lambda}|) \\ + 2\mathbf{tr}(\tilde{\mathbf{K}}_r^T \boldsymbol{\Gamma}_r^{-1} \dot{\tilde{\mathbf{K}}}_r |\boldsymbol{\Lambda}|)\end{aligned}$$

$$\begin{aligned}= \mathbf{e}^T(\mathbf{P}\,\mathbf{A}_m + \mathbf{A}_m^T\mathbf{P})\mathbf{e} + 2\mathbf{e}^T\mathbf{P}\mathbf{B}\boldsymbol{\Lambda}\left(\tilde{\mathbf{K}}_y^T\mathbf{y} + \tilde{\mathbf{K}}_r^T\mathbf{r} - \boldsymbol{\varepsilon}_f(\mathbf{x})\right) \\ +2\mathbf{tr}\left(\tilde{\mathbf{K}}_y^T \boldsymbol{\Gamma}_{\boldsymbol{y}}^{-1} \dot{\tilde{\mathbf{K}}}_y |\boldsymbol{\Lambda}|\right) + 2\mathbf{tr}\left(\tilde{\mathbf{K}}_r^T \boldsymbol{\Gamma}_r^{-1} \dot{\tilde{\mathbf{K}}}_r |\boldsymbol{\Lambda}|\right)\end{aligned}$$

$$\begin{aligned}= -\mathbf{e}^T\mathbf{Q}\mathbf{e} - 2\mathbf{e}^T\mathbf{P}\mathbf{B}\boldsymbol{\Lambda}\boldsymbol{\varepsilon}_f(\mathbf{x}) + 2\mathbf{e}^T\mathbf{P}\mathbf{B}\boldsymbol{\Lambda}\tilde{\mathbf{K}}_y^T\mathbf{x} \\ + 2\,\mathbf{tr}\left(\tilde{\mathbf{K}}_y^T \boldsymbol{\Gamma}_{\boldsymbol{y}}^{-1} \dot{\tilde{\mathbf{K}}}_y\right) + 2\mathbf{e}^T\mathbf{P}\mathbf{B}\boldsymbol{\Lambda}\tilde{\mathbf{K}}_r^T\mathbf{r} + 2\,\mathbf{tr}\left(\Delta\mathbf{K}_r^T \boldsymbol{\Gamma}_r^{-1} \dot{\tilde{\mathbf{K}}}_r\right)\end{aligned}$$

Since $x^T\,y = \mathbf{tr}\left(y\,x^T\right)$ from trace identity, we have

$$\begin{aligned}\dot{\mathbf{V}}(\cdot) = -\mathbf{e}^T\mathbf{Q}\mathbf{e} - 2\mathbf{e}^T\mathbf{P}\mathbf{B}\boldsymbol{\Lambda}\boldsymbol{\varepsilon}_f \\ + 2\,\mathbf{tr}\left(\tilde{\mathbf{K}}_y^T(\boldsymbol{\Gamma}_{\boldsymbol{y}}^{-1} \dot{\tilde{\mathbf{K}}}_y + \mathbf{x}\mathbf{e}^T\mathbf{P}\mathbf{B}sgn(\boldsymbol{\Lambda})\right)|\boldsymbol{\Lambda}| \\ + 2\,\mathbf{tr}\left(\tilde{\mathbf{K}}_r^T(\boldsymbol{\Gamma}_r^{-1} \dot{\tilde{\mathbf{K}}}_r + \mathbf{r}\mathbf{e}^T\mathbf{P}\mathbf{B}sgn(\boldsymbol{\Lambda})\right)|\boldsymbol{\Lambda}|\end{aligned} \tag{4.35}$$

where for a real-valued x, we have $x = \text{sgn}(x)|x|$. The first two terms in (4.35) will be negative definite for all $\mathbf{e} \neq 0$ since $\mathbf{A}_m$ is Hurwitz and the other terms in (4.35) will be identically null if we choose the adaptation laws

$$\dot{\hat{\mathbf{K}}}_y = -\mathbf{\Gamma}_{\boldsymbol{y}}\mathbf{y}\mathbf{e}^T\mathbf{P}\,\mathbf{B}\text{sgn}(\mathbf{\Lambda}), \dot{\hat{\mathbf{K}}}_r = -\mathbf{\Gamma}_r\mathbf{r}\mathbf{e}^T\mathbf{P}\,\mathbf{B}\,\text{sgn}(\mathbf{\Lambda}). \tag{4.36}$$

The time-derivative of the Lyapunov function can then be written as

$$\begin{aligned}\dot{\mathbf{V}}(\cdot) &= -\mathbf{e}^T\mathbf{Q}\mathbf{e} - 2\mathbf{e}^T\mathbf{P}\mathbf{B}\mathbf{\Lambda}\boldsymbol{\varepsilon}_f \\ &\leq -\lambda_{low}\|\mathbf{e}\|^2 + 2\|\mathbf{e}\|\|\mathbf{P}\mathbf{B}\|\lambda_{high}(\mathbf{\Lambda})\boldsymbol{\varepsilon}_{max},\end{aligned} \tag{4.37}$$

where $\lambda_{low}, \lambda_{high}$ represent the minimum and maximum characteristic roots of Q and $\mathbf{\Lambda}$ respectively. $\dot{\mathbf{V}}(\cdot)$ is thus negative definite outside the compact set

$$\chi = \left(\mathbf{e} : \|\mathbf{e}\| \leq \frac{2\|\mathbf{P}\mathbf{B}\|\lambda_{high}(\mathbf{\Lambda})\varepsilon_{max}}{\lambda_{low}(\mathbf{Q})}\right). \tag{4.38}$$

and we conclude that the error $\mathbf{e}$ is *uniformly ultimately bounded*. As $\mathbf{e}$ converges to a neighborhood of 0, $\mathbf{y}$ converges to a neighborhood of $\mathbf{y}_m$. From the stable model reference system in (4.24), $\mathbf{y}$ converges to a neighborhood of $\mathbf{r}$. Note that asymptotic convergence of $\mathbf{e}$ to zero is not guaranteed but the parametric errors are guaranteed to stay bounded.

4.3.5 Network Design

We choose a LSTM (Hochreiter and Schmidhuber, 1997) due to its capacity for long-term context memorization and inherent multiplicative units that avoid oscillating weights or vanishing gradients when error signals are backpropagated in time (Bengio et al., 1994; Hochreiter and Schmidhuber, 1997). LSTMs truncate gradients in the network where it is harmless by enforcing constant error flows through their *constant error carousels*. As a result, LSTMs are robustly more powerful for adaptive sequence-to-sequence modeling or mapping data that temporally evolve in time. Their biological model makes them more suitable for adaptive robotics such as soft robots than

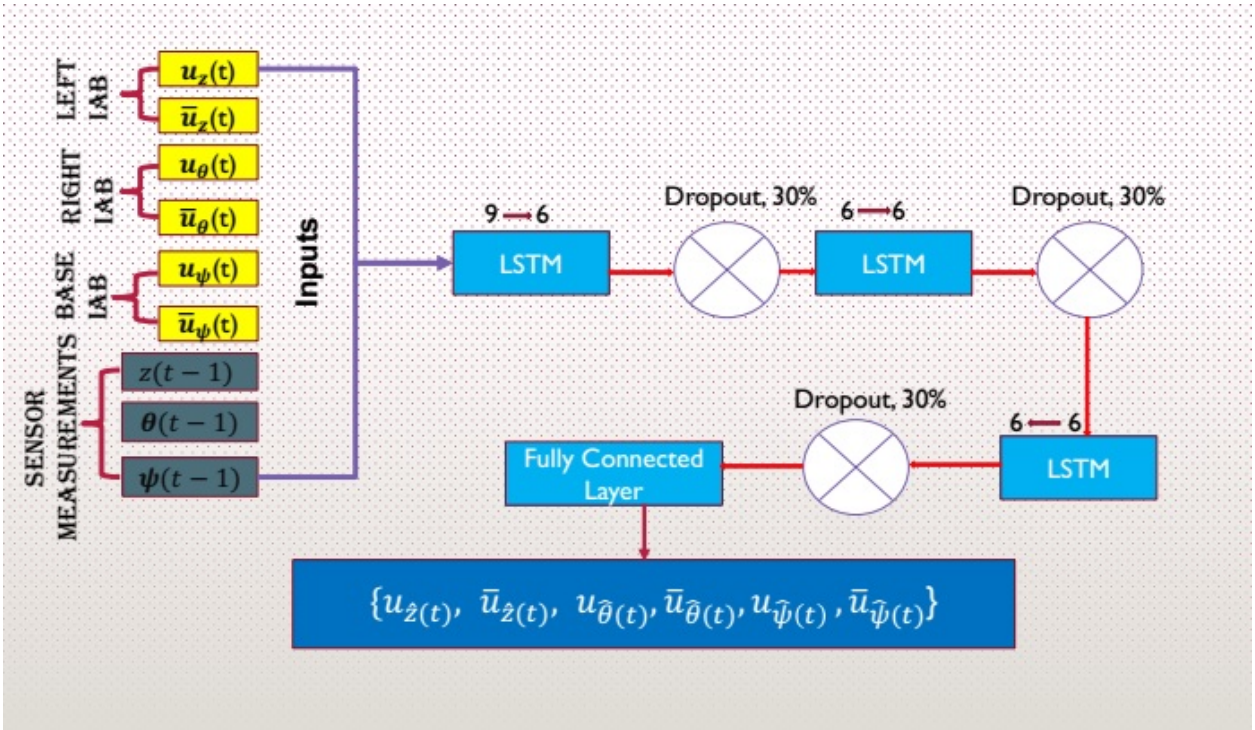


Figure 4.10: Neural Network model

previously used artificial NNs such as feedforward networks (Dinh et al., 2017), radial basis-functions (Lavretsky and Wise, 2005; Patino and Liu, 2000) or vanilla RNNs (Wang and Chen, 2006).

The neural network is shown in Figure 4.10. Depending on the region of attraction of the system the network is approximating, it parameterizes the nonlinear dynamical system $f(\cdot)$ and maps the parameterized model to appropriate valve torques. There exists additional feedforward + feedback terms in the global controller (introduced shortly) that guarantee system stability and robustness to uncertainties. Therefore, the global controller keeps the states of the system bounded under closed-loop dynamics, ensures convergence to desired trajectories from states that are initialized outside the domain of attraction, and guarantees robust reference tracking in the presence of non-parametric uncertainties.

The NN model takes a memory-based concatenated vector of current inputs and past outputs as in (4.26), propagates them through three hidden layers, with each layer made up of $\{9, 6, 6\}$

neurons each, applies 30% dropout and then maps the last layer to a fully connected layer that generates valve torques. A self-tuning adaptive control law (with a feedforward regulation and state feedback component) adapts to the internal parameters of the plant to ensure stability of the system and bounded tracking of given trajectory. The overall network has neuron connection weights and thresholds of approximately 1,400. This makes search for a suitable controller feasible.

The LSTM network estimates a model, $f(\mathbf{x})$, that minimizes the mean-squared error between predicted output, $\hat{\mathbf{x}}(k)$, and actual output $\mathbf{x}(k)$ according to

$$f(\mathbf{x}(k)) = \arg\ \min_{w}\ \mathbf{V}_N(w, \mathbf{\Phi}(\mathbf{x})) \tag{4.39}$$

where $\mathbf{V}_N(\mathbf{w}, \mathbf{\Phi}(\mathbf{x})) = \sum_{t=1}^{\mathcal{K}} \sum_{i=1}^{n} \frac{1}{2}(\hat{\mathbf{x}}_i(t) - \mathbf{x}_i(t))^2$, and $\Phi(\mathbf{x})$ is a regression vector as defined in (4.26) on a bounded interval $[1, N]$. (4.39) is minimized using stochastic gradient descent so that at each iteration, we update the parameters (weights) of the network $\mathbf{w}_i$ based on the *ordered derivatives* of $\mathbf{V}_N(\mathbf{w}, \mathbf{\Phi}(\mathbf{x}))$ (Werbos (Werbos, 1990)) $i.e.$

$$\mathbf{w}_{k+1} \leftarrow \eta \mathbf{w}_k - \alpha \sum_{i=1}^{n} \nabla_{\mathbf{w}} \mathbf{V}(\mathbf{x}_i, \hat{\mathbf{x}}_i(\theta_k)). \tag{4.40}$$

η (set to 1) hastens the optimization in a direction of low but steepest descent in training error, and α is a sufficiently small learning rate (set to 5×10^{-3}), and $\nabla_w \mathbf{V}(\theta, \mathbf{\Phi}(\mathbf{x}))$ is the derivative of $\mathbf{V}$ with respect to w averaged over the k-th batch (we used a batch size of 50).

4.3.6 Results and Discussion

We sample from the parameters of the trained network and we set $\hat{f}(\cdot)$ in (4.25) to the fully connected layer of samples from the network. We publish the control law from the neural network and subscribe in a separate node. The gains $\hat{\mathbf{K}}_y$ and $\hat{\mathbf{K}}_r$ were found by solving the ODEs iteratively using a single step of the integral of the solutions to $\dot{\hat{\mathbf{K}}}_y(t)$, $\dot{\hat{\mathbf{K}}}_r(t)$. Our solution is an implementation of the Runge-Kutta Dormand-Prince 5 ODE-solver available in the Boost C++ Libraries[4]. We found

[4]https://goo.gl/l7JyYe

a step-size of 0.01 to be realistic. $\mathbf{x}_m$ in (4.24) is computed based on the solution to the forced response of the linear system,

$$\mathbf{y}_m(t) = e^{\mathbf{A}_m t}\mathbf{y}_m(0) + \int_0^t e^{\mathbf{A}_m(t-\tau)}\mathbf{B}_m\,\mathbf{r}(\tau)d\tau.$$

For a nonnegative $\mathbf{Q}$ and a positive definite $\mathbf{P}$, the pair $(\mathbf{Q}, \mathbf{A}_m)$ will be observable so that the dynamical system is globally asymptotically stable. After searching, we picked a positive definite $\mathbf{Q} = diag(100, 100, 100)$ for the dissipation energy and set $\mathbf{\Lambda} = I_{3\times 3}$ so that solving the general form of the lyapunov equation, we have P. The six solenoid valves operate in pairs so that two valves create a difference in air mass within each IAB at any given time. Therefore, we have

$$\mathbf{P} = \begin{bmatrix} -\frac{170500}{2668} & 0 & 0 \\ 0 & -\frac{170500}{2668} & 0 \\ 0 & 0 & -\frac{170500}{2668} \end{bmatrix}, . \quad \mathbf{B} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \tag{4.41}$$

The non-zero terms for B in (4.41) are the maximum duty-cycle that can be applied to the solenoid valves based on the software configuration of the NI RIO PWM generator.

The three DOFs of the head are coupled and there is a limited reachable space with the IABs. It is therefore paramount that desired trajectories be ascertained as physically realizable before rolling out control trials. We therefore placed the head to physically realizable positions in open-loop control settings before testing the close-loop control system.

Figure 4.11 show the performance of the controller when commanded to move the head from $[z, \theta, \phi]^T = [2.5mm, .25^o, 35^o]^T$ to $[14mm, 1.6^o, 45^o]^T$. We observe strong steady-state convergence along 2-DOFs , namely z and pitch axes with a $20sec$ rise time. The right portion of the figure demonstrates the roll angle tracking by the head. The roll-motion is well-conditioned with minimal disturbance around the set-point.

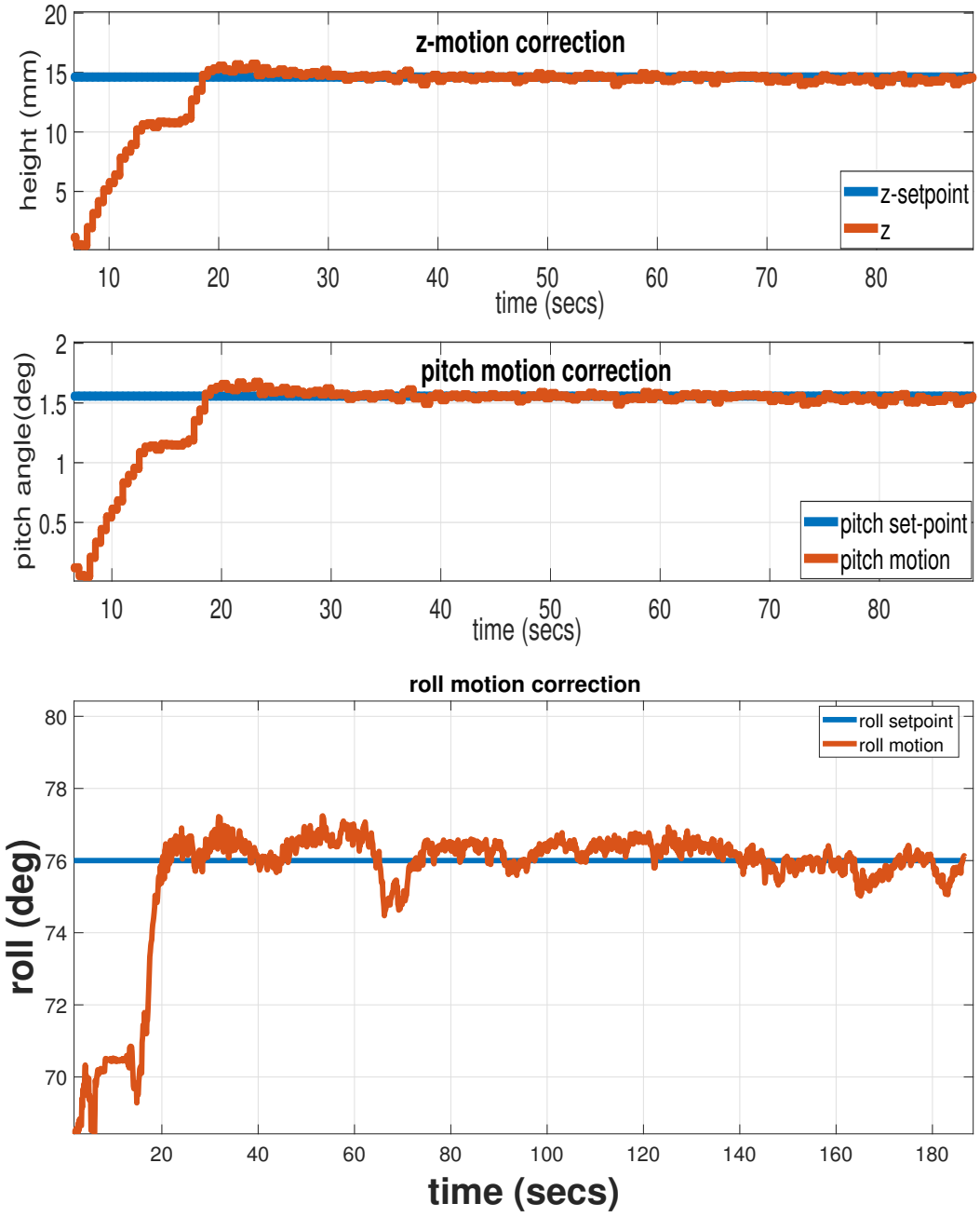


Figure 4.11: Head motion correction along z, pitch and roll axes.

CHAPTER 5

AUTOMATING BEAM ORIENTATION OPTIMIZATION

In this chapter, we present our proposed algorithm for solving the BOO problem in IMRT treatment planning. We treat the problem as a suboptimal control problem within the framework of approximate dynamic programming (ADP) (Bertsekas et al., 2017a,b). Particularly, we leverage recent machine learning breakthroughs (LeCun et al., 2015) to guide a Monte Carlo Tree Search (MCTS) (Coulom, 2006; Gelly and Silver, 2011; Kocsis and Szepesvári, 2006) of promising beam angle candidates by rapidly exploring different parts of the beam space. Our goal is to mitigate the time it takes for clinicians to find suitable beam angles in intensity-modulated radiation therapy (IMRT). As we remarked in § 1.4, the beam orientation problem is difficult to crack given the non-convexity of the solution surface, the heavy computational requirement for the search problem, and the way a minor change in beam orientation alters new dose intensity distributions. When just the tip point of the robot's end effector rotates the *gantry* that contains the linear accelerator, we have **coplanar beams** (see Figure 5.1. Our focus is on coplanar beams, given that only coplanar beams are employed (Aleman et al., 2008b) in IMRT treatment procedures. We consider fictitious self-play as a practical application for developing an effective beam orientation selection strategy in a scenario involving two rational decision-making agents that: i) do not communicate their policies to each other (i.e. the game is non-cooperative), and ii) behave reactively in order to adequately explore the state space. Thus, aiming to automate the beam orientation selection and intensity modulation process, we introduce a novel set of techniques leveraging (i) pattern recognition, (ii) Monte Carlo evaluations, (iii) game theory, and (iv) neuro-dynamic programming. A deep neural network policy guides Monte Carlo simulations of promising beamlets. Seeking a saddle equilibrium, two fictitious neural network players, within a zero-sum Markov game, alternatingly play a best response to an opponent's mixed strategy profile during episodes of a two-player Markov decision game. After a sufficient number of games of self-play, the optimized policy predicts beam angles on test target

Figure 5.1: IMRT TPS setup. Reprinted from Radiologyinfo.

volumes. We merge the beam orientation search and the subsequent fluence map optimization subproblems.

5.1 Methods and Materials

Consider b^d possible move sequences of a setup, where b are the beam angles chosen to construct a fluence, and d is the total number of discretized beam angles within the beams space. Suppose $b = 180$ and $d = 5$, we have 180^5 possible search directions, rendering search computationally demanding. Finding the optimal solution would require a tiring search. Monte Carlo simulations have been successful at breaking the curse of dimensionality (Bellman, 1957) in decision-making in games with large state spaces such as Go, and Shogi, inter alia (Chung et al., 2005; Silver et al., 2016, 2017). For games with perfect information, there is an optimal value function, $v^\star(s)$, that decides the game's outcome for every possible state, $s \in \mathcal{S}$, under perfect play ($i.e.$ a game with an

optimal min-max state-action value function, $Q(s, u)$, produced by a policy $\boldsymbol{\pi} = \{\boldsymbol{\pi}_{min}, \boldsymbol{\pi}_{max}\}$). Our task is to devise a planning strategy that guides the search for optimistic beam angles within the setup's phase space using a probability distribution, $p(s, u)$, over a set of deterministic *pure strategies* for the tree. The pure strategies in this case would be probability distributions over the space of every possible beam angle index within the available beams set, u, at time step, t.

The search for an *approximately* optimal beam angle set is performed by optimizing the parameters of a function approximator ψ, (here, a deep neural network, with multiple residual blocks (He et al., 2016)); this network approximates the policy, $\boldsymbol{\pi}$, which guides MCTS simulations of 'best-first' beam angle combinations for a sufficiently large number of iterations. This MCTS performs a sparse lookout simulation, recursively expanding child nodes of a tree, while selectively adjusting beam angles that contribute the least to an optimal fluence profile. Successor nodes beneath a terminal node are **approximated** with a value, $v(s)$, to assure efficient selectivity. We maintain a probability distribution over possible states (beam angles), based on a set of observation probabilities for the underlying Markov Decision Process (MDP).

5.1.1 Notations and Definitions

The state of the dynamical system will be denoted by $\mathbf{x} \in \mathcal{X}$; it is to be controlled by a discrete action $u \in \mathcal{U}$. States evolve according to the (unknown) dynamics $p(\mathbf{x}_{t+1}|\mathbf{x}_t, u_t)$, which we want to learn. The learning problem is posed within a discrete finite-time horizon, T, while a beam angle combination search task can be defined by a reward function, $r(\mathbf{x}_t, u_t)$, to be found by recovering the policy, $p(u_t|\mathbf{x}_t; \psi)$, that specifies distribution over actions conditioned on the state, and parameterized by the weights of a neural network, a tensor ψ. Without loss of generality, we denote the action conditional $p(u_t|\mathbf{x}_t, \psi)$ as $\boldsymbol{\pi}_\psi(u_t|\mathbf{x}_t)$. Recovering the optimal weights may consist of the maximization problem

$$\psi^\star = \arg\max_\psi \sum_{t=1}^{T} \mathbb{E}_{(\mathbf{x}_t, u_t)\sim p(\mathbf{x}_t, u_t|\psi)} \left[r(\mathbf{x}_t, u_t) \right].$$

Table 5.1: Table of Notations.

Notation	Definition/Examples	Notation	Definition/Examples
m	dimensionality of a node's beam set, e.g. $m = 5$	n	dimension of discretized beam angles, e.g. $n = 180$ for 4° angular resoultion
$\boldsymbol{\Theta}$	discretized beam angle set e.g. equally spaced angles between 0° and 360°, spaced apart at 4°	$u_t \in \mathcal{U}$	control or action, $u_t \in \mathcal{U}$ at time step $t \in [1, T]$ used in determining the probability of transitioning from a beam angle subset to another within $\boldsymbol{\Theta}$
$\theta^j \subseteq \boldsymbol{\Theta}$	beam angles selected from $\boldsymbol{\Theta}$ e.g. $\theta_k \in \mathbb{R}^m$	$\mathbf{x}_t \in \mathcal{X}$	Markovian system state at time step, $t \in [1, T]$ e.g. patient contour, beam angle candidates; dimensionality $2,727,936$ to $3,563,520$
γ	discount factor e.g. 0.99	$\boldsymbol{f_\psi}$	parametric function approximator (deep neural network policy) for state $\mathbf{x}_t$
$v_\psi(\mathbf{x})$	value estimate of state, $\mathbf{x}_t$, as predicted by $\boldsymbol{f_\psi}$	$\boldsymbol{p}(\mathbf{x})$	probability distribution over current state, $\mathbf{x}$ generated by neural network policy
$Q(\mathbf{x}, u)$	action-state values that encode the "goodness" of a beam-angle set, $\theta_k \in \mathbb{R}^m$, where m is the number of beams considered for a fluence generation, e.g. $m = 5$	$\boldsymbol{B}_{X_t}$	a concatenation of beams in consideration at time step, t, as a block of beams being fed to the neural network policy
$\mathcal{D}_{ij}(\theta_k)$	dose matrix containing dose influence to voxel i from beam angle, θ_k, $\forall\, k \in \{1, 2, \ldots, n\}$ where n is range of the beam set $\mathcal{B}$	$\boldsymbol{D}_t$	dose mask for target volume in consideration at time step , t

where the weight tensor $\psi^\star$ maximize the sum of the respective instantaneous rewards $\sum_t r(\mathbf{x}_t, \boldsymbol{u}_t)$ of the policy, $\boldsymbol{\pi}_\psi(u_t|\mathbf{x}_t)$. Let us now introduce some definitions.

Definition 1. A **beam block** is a concatenation of beams, $\{\theta_1, \theta_2, \ldots, \theta_m\}$ as a tensor of dimension $m \times N \times H \times W$, (see Figure 5.3 and Table 5.1) where N is the total number of slices, W is the slice width, and H is the slice height, that together with the patient's ct mask form the state, $\mathbf{x}_t$, at time step, t.

Other notations are delineated in Table 5.1.

Definition 2. Every **node** of the tree, $\boldsymbol{x}$, has the following fields:

- a pointer to the parent that led to it, $\boldsymbol{x}.p$;

(a) angle 10° (b) angle 166° (c) angle 286° (d) angle 342°

Figure 5.2: Example 2D gantry angle representations

- the beamlets, $\boldsymbol{x}_b$, stored at that node where $b = \{1, \ldots, m\}$, where m is the number of beam plans being considered for the irradiation problem, e.g. $m = 5$;
- a set of move probabilities prior, $p(s, \boldsymbol{u})$;
- a pointer, $\boldsymbol{x}.r$, to the reward, r_t, for the state s_t;
- a pointer to the state-action value $Q(s, \boldsymbol{u})$ and its upper confidence bound $U(s, \boldsymbol{u})$ (to be shortly introduced)
- a visit count, $\mathbb{N}(s, \boldsymbol{u})$, that indicates the number of times that node was visited; and
- a pointer $\boldsymbol{x}.child_i$ to each of its children nodes.

5.1.2 Data Preprocessing

We obtained 77 anonymized patient CT scans from our clinic and their associated dose matrices. The scans relate to prostate cases used in previous IMRT treatment planning. The prostate data contains the scan of six organs, namely the patients' body, bladder, left and right femoral heads, rectum, and planning target volume (PTV) or tumor. Each patient's scan, $\boldsymbol{D}$, is represented in 3D as $N \times W \times H$. We resized each slice to a square-shaped 2D matrix of size 64×64. We generate 3D images that represent the orientation of the robot with respect to the patient for each discretized beam angle as illustrated in Figure 5.2.

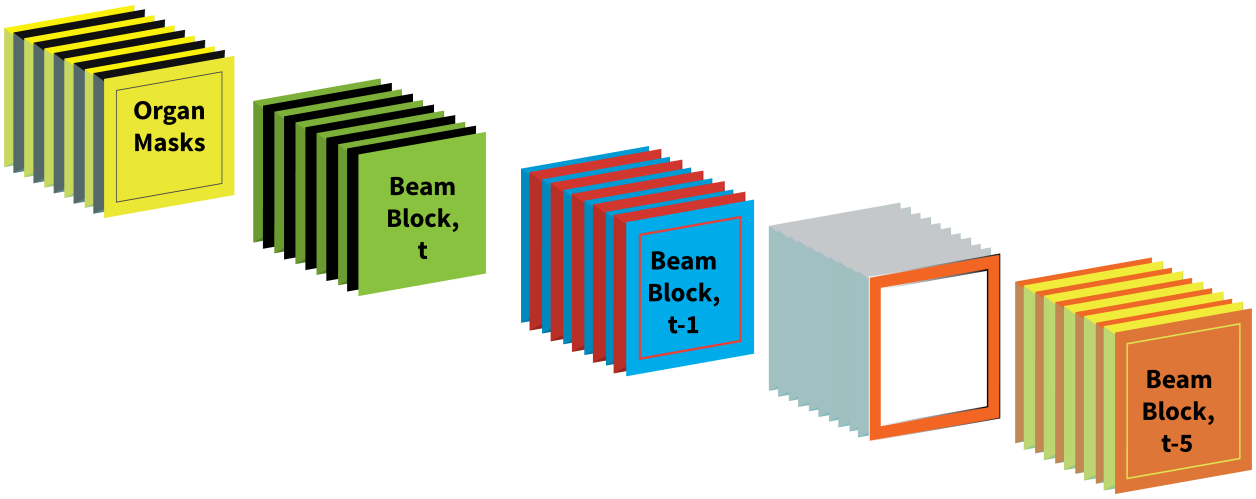


Figure 5.3: Network's input planes.

5.1.3 Neural Network Architecture

In addition to the resized masks, $\boldsymbol{D}$, we define five feature planes, $\boldsymbol{X}_t$, as beam configuration blocks: $\boldsymbol{B}_{\boldsymbol{X}_t}$. $\boldsymbol{B}_{\boldsymbol{X}_t}$ denotes the beam angles that generate the current fluence. For five beams for the fluence's geometric shape for example, $\boldsymbol{B}_{\boldsymbol{X}_t}$ would contain the RGB images of the beams being considered at time step t. We augment the state with a memory of five previously used beam blocks, $\{\boldsymbol{B}_{\boldsymbol{X}_t}, \ldots, \boldsymbol{B}_{\boldsymbol{X}_{t-5}}\}$. The input planes to the network are sized as $T \times N \times H \times W$, where T is the total number of input planes ($T = 6$ structures $+ 5$ beams $+ 5 \times 5$ regressed beams $= 36$). Thus, the input to the network are arranged as: $\mathbf{x}_t = [\boldsymbol{D}_t, \boldsymbol{B}_{\boldsymbol{X}_t}, \boldsymbol{B}_{\boldsymbol{X}_{t-1}}, \boldsymbol{B}_{\boldsymbol{X}_{t-2}}, \boldsymbol{B}_{\boldsymbol{X}_{t-3}}, \boldsymbol{B}_{\boldsymbol{X}_{t-4}}, \boldsymbol{B}_{\boldsymbol{X}_{t-5}}]$. The dose masks and beam blocks are as illustrated in Figures 5.3 and 5.4.

A tower residual network (He et al., 2016) fits residual nonlinear mappings to the input planes so that we end up with a deeply stacked network whose input features, $\mathbf{x}_t$, are processed by 34 residual blocks described as follows: (i) a 3D convolution with $64 \times l$ filters, a square kernel of width 7, and double strided convolutions, where l is the depth of the stack in the network; (ii) a 3D batch normalization layer (Ioffe and Szegedy, 2015); (iii) nonlinear rectifiers (Hahnloser et al., 2000);

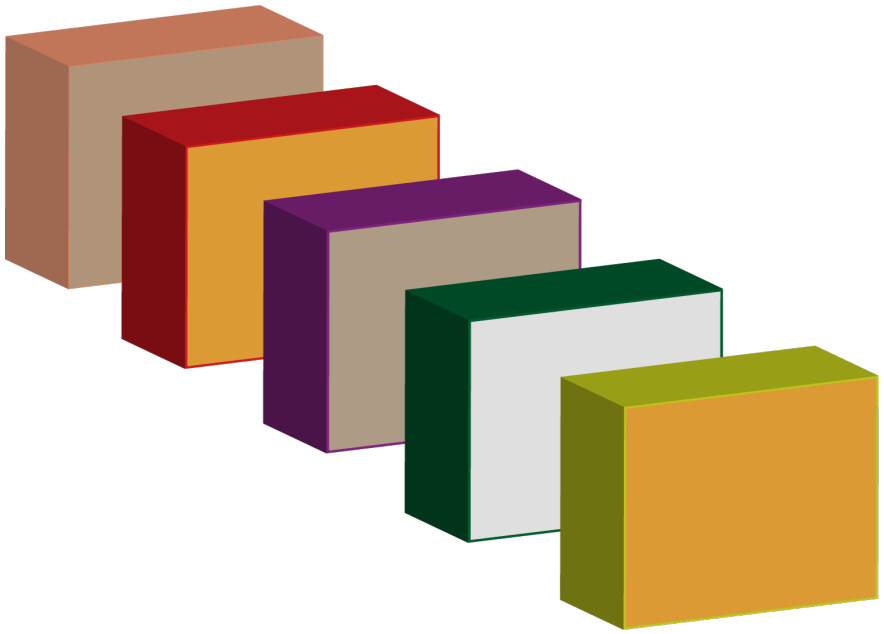


Figure 5.4: Each beam angle in a beam block is represented as shown. Together with the target volume, these form an input plane of size $36 \times N \times W \times H$ to the policy/value neural network tower of residual blocks.

(iv) a 3D convolution of $64 \times l$ filters; (v) a 3D batch normalization layer; (vi) a skip connection from the input to the block, in order to facilitate efficient gradients' propagation; and (vii) nonlinear rectifiers.

The output of the network divides into two heads: (i) the first head is a probability distribution over which angle in the current beam block contributes the least to an optimal fluence cost at the current time step, while (ii) the second head estimates the *value* (in an *ADP sense*) of the subtree beneath the current node. The network's parameters were initialized using the proposal in (He et al., 2015). The value and probability distribution heads are inspired from Bayesian decision theory, where it is expected that a rational decision-maker's behavior is describable by a *utility function*, (or value function) – a quantitative characterization of the policy's preferences for an outcome – and a subjective probability distribution, which describes the policy's beliefs about all relevant unknown factors. When new information is presented to the decision-maker, the subjective probability distribution gets revised. Decisions about the optimal beam angle combination at the current time step are made under uncertainty; so we use a *probability model* to choose among

lotteries, ($i.e.$ probability distributions over all discretized beam angles in the setup). Each state during our learning process is constructed by appending the beam block at the current time step to a history of beam blocks for the previous five time steps using a FIFO policy. Specifically, when we transition to a new state, the beam block that has been in the state set for the longest time ($i.e.$ at the *head* of the queue) is deleted first, and the new state's beam block is enqueued at the tail as in a *queue* data structure. This is so as to minimize the partial observability of the system.

5.1.4 Fluence Map Optimization

Suppose $\mathcal{X}$ is the total number of discretized voxels of interest (VOIs) in a target volume, and $\mathcal{B}_1 \cup \mathcal{B}_2 \cup \ldots \cup \mathcal{B}_n \subseteq \mathcal{B}$ represents the partition subset of a beam $\mathcal{B}$, where n is the total number of beams from which radiation can be delivered. Let $\mathcal{D}_{ij}(\theta_k)$ be the matrix that describes each dose influence, d_i, delivered to a discretized voxel, i, in a volume of interest, VOI_h $(h = 1, \ldots, \mathcal{X})$, from a beam angle, θ_k, $k \in \{1, \ldots, n\}$. One can compute the matrix $\mathcal{D}_{ij}(\theta_k)$ by calculating each d_i for every bixel, j, at every φ°, resolution, where $j \in \mathcal{B}_k$. Doing this, we end up with an ill-conditioned *sparse* matrix, $\mathcal{D}_{ij}(\theta_k)$, which consists of the dose to every voxel, i, incident from a beam angle, θ_k at every $360^\circ/\varphi^\circ$ (in our implementation, we set φ to 4°).

For a decision variable, x_j, representing the intensities of beamlets, it is trivial to find the dose influence, d_i, that relates the bixel intensities, x_j, to the voxels of interest, VOI_h. The fluence problem is to find the values of x_j for which d_i to the tumor is maximized, while simultaneously minimizing the d_i to critical structures. For the voxels in the target volume, a weighted quadratic objective minimizes the l_2 distance between a pre-calculated dose $\boldsymbol{Ax}$, and a doctor's prescribed dose, $\mathbf{b}$, while a weighted quadratic objective maximizes the l_2 distance between $\boldsymbol{Ax}$ (where $\boldsymbol{x}$ represents the vectorized bixels, x_j) and $\mathbf{b}$. The pre-calculated dose term is given by $\boldsymbol{Ax} = \{\sum_s \frac{w_s}{v_s} \mathcal{D}_{ij}^s \boldsymbol{x}_s \,|\, \mathcal{D}_{ij} \in \mathbb{R}^{n \times l}, n > l\}$, which is a combination of the dose components that belong to OARs and those that belong to PTVs. Let $w_s = \{\underline{w}_s, \bar{w}_s\}$ represent the respective underdosing and overdosing weights for the OARs and PTVs, and v_s represents the total number of voxels in each

structure. The cost function is

$$\frac{1}{v_s}\sum_{s\in\text{OARs}}\|(b_s-\underline{w}_s\mathcal{D}_{ij}^s\boldsymbol{x}_s)_+\|_2^2+\frac{1}{v_s}\sum_{s\in\text{PTVs}}\|(\bar{w}_s\mathcal{D}_{ij}^s\boldsymbol{x}_s-b_s)_+\|_2^2 \tag{5.1}$$

where the underdosing weights are typically set as $\underline{w}_s = 0$ to deliver minimal dose to critical structures, while the overdosing weights are chosen to deliver the prescribed dose to the tumor; $(\cdot)_+$ denotes a Euclidean projection onto the nonnegative orthant $\mathbb{R}_+$. We solve (5.1) with the alternating direction method of multipliers (ADMM) (Boyd et al., 2011) algorithm by adding a quadratic penalty term to the Lagrangian of (5.1) and alternatingly updating the $\boldsymbol{x}$ and $\boldsymbol{\lambda}$ variables in a "broadcast and gather" process. This turns out to be attractive, since we will be solving a large-scale learning problem for the optimal beam angle set combination. The ADMM update equations (where **z** is the ADMM dual variable) is

$$\begin{aligned}\boldsymbol{x}^{k+1}&=\left(\boldsymbol{A}^T\boldsymbol{A}+\rho\boldsymbol{I}\right)^{-1}\left(\boldsymbol{A}^T\boldsymbol{b}+\rho\boldsymbol{z}^k-\boldsymbol{\lambda}^k\right)\\ \boldsymbol{z}^{k+1}&=S_{\boldsymbol{\lambda}/\rho}\left(\boldsymbol{x}^{k+1}+\boldsymbol{\lambda}^k\right)\end{aligned} \tag{5.2}$$

where $S_{\boldsymbol{\lambda}/\rho}(\tau)=(x-\boldsymbol{\lambda}/\rho)_+-(-\tau-\boldsymbol{\lambda}/\rho)_+$, $\boldsymbol{\lambda}$ is updated as

$$\boldsymbol{\lambda}^{k+1}=\boldsymbol{\lambda}^k-\gamma(\boldsymbol{z}^{k+1}-\boldsymbol{x}^{k+1}), \tag{5.3}$$

and γ is a parameter that controls the step length. We refer readers to Appendix D for these derivations.

5.2 Game Tree Simulation

A *game* may refer to an event involving two or more agents, often called *players*. Interactive decision-making among players in a game allows us to establish a *utility function*, which gives the quantitative analysis of a player's predilection for certain outcomes, and a *subjective probability distribution*, which portrays important unknown factors. We introduce a set of basic game notations and definitions to guide our exposition. A game has *perfect information* if no two nodes have the

same information state, where an *information state* specifies the state that a player would have if the path reached that node(Roger, 1991). For a class of two-player zero-sum games of perfect information, a state is called a *position* while the action is referred to as a *move*. In *fictitious self-play (FSP)* games, each player iteratively plays a *best response* to their opponent's *average strategy* in a repeated fashion. For two-player fictitious games, the average strategy profile of the players converge to a *Nash equilibrium* (Heinrich et al., 2015). The finite set of *legal actions* in *state* $\mathbf{x} \in \mathcal{S}$ is denoted as $\mathcal{U}(\mathbf{x})$. A game is complete when it reaches a *terminal state* with an *outcome* ζ.

The *security strategy* of each player in a game is the strategy among all available alternatives that guarantees that each player's loss after an opponent has played is no greater than its *loss ceiling* or each player's gain is no smaller than its *gain floor* ($i.e.$ their *security level*). Security strategies are in equilibrium when one player's strategy is optimal against that of another. Two strategies are said to be in *equilibrium* if after finishing a game, and the outcome is observed, the players do not have reasons to regret actions they took in the past. When the security levels of the players coincide, the *outcome* of a game will always be the same and such equilibrium strategies are said to be *saddle-point strategies* and the game is said to have a *saddle point in pure strategies* (Basar and Olsder, 1999).

In this section, we continually play a zero-sum FSP game between two neural networks. Without loss of generality, we will call the first player the nominal player, or player p_1, and the second player the adversarial player, or player p_2. Player p_1 chooses its action under a (stochastic) strategy, $\pi^{p_1} = \{\pi_0^{p_1}, \pi_1^{p_1}, \ldots, \pi_T^{p_1}\} \subseteq \Pi^{p_1}$ that seeks to minimize the outcome ζ, while p_2's actions are governed by a policy $\pi^{p_2} = \{\pi_0^{p_2}, \pi_1^{p_2}, \ldots, \pi_T^{p_2}\} \subseteq \Pi^{p_2}$ that seeks to maximize ζ in order to guarantee an equilibrium solution for a game without saddle point. Π^{p_i} is the set of all possible nonstationary Markovian policies. Each player bases its decision on a random event's outcome – generating a *mixed strategy* determined by averaging the outcome of individual plays. Together, both players constitute a two-player *stochastic action selection strategy*, $\boldsymbol{\pi}(\mathbf{x}, u) = Pr(u|\mathbf{x}) := \{\boldsymbol{\pi}^{p_1}, \boldsymbol{\pi}^{p_2}\}$ that gives the probability of selecting moves in any given state. Suppose the game simulation starts from

an initial condition $\mathbf{x}_0$, one may write the optimal reward-to-go value function for state s in stage t, with horizon length T as

$$V_t^*(\mathbf{x}) = \inf_{\pi^{p_1} \in \Pi^{p_1}} \sup_{\pi^{p_2} \in \Pi^{p_2}} \mathbb{E}\left[\sum_{i=t}^{T-1} V_t(\mathbf{x}_0, f(\mathbf{x}_t, \pi^{p_1}, \pi^{p_2}))\right],$$

$$\mathbf{x} \in S,\, t = 0, \ldots, H-1$$

where the terminal value $V_T^*(\mathbf{x}) = 0,\ \forall\, s \in S;\ f(\cdot)$ represents the unknown system dynamics, π^{p_1} and π^{p_2} contain the action/control sequences $\{u_t^{p_1}\}_{0 \le t \le T}$ and $\{u_t^{p_2}\}_{0 \le t \le T}$. The *saddle point strategies* for an optimal control sequence pair $\{u_t^{p_1^*}, u_t^{p_2^*}\}$ can be recursively obtained by optimizing a state-action value cost, $\mathcal{J}_t(\mathbf{x}, u)$ as follows

$$V_t^*(\mathbf{x}) = \mathcal{J}_t^*(\mathbf{x}_t, \pi_t^{p_1}, \pi_t^{p_2}) = \min_{\pi^{p_1} \in \Pi^{p_1}} \max_{\pi^{p_2} \in \Pi^{p_2}} \mathcal{J}_t^\star(\mathbf{x}_t, \pi^{p_1}, \pi^{p_2})$$

$$\forall \mathbf{x}_t \in \mathcal{S}, \pi^{p_1} \in \Pi^{p_1}, \pi^{p_2} \in \Pi^{p_2}. \tag{5.4}$$

such that

$$v_{p_1}^\star \le v^\star \le v_{p_2}^\star \quad \forall \left\{\pi_t^{p_1}, \pi_t^{p_2}\right\}_{0 \le t \le T}.$$

where $v_{p_i}^\star$ are the respective optimal values for each player. $Q(\mathbf{x}, u)$ can be recovered from the reward function $R(\mathbf{x}, u)$ and transition function, $P(\mathbf{x}, u)$ as

$$\mathcal{J}_t^*(\mathbf{x}_t, \pi_t^{p_1}, \pi_t^{p_2}) = R(\mathbf{x}, u) + \gamma \sum_{x \in S} P(\mathbf{x}, u)(x) V_{t+1}^\star(x).$$

The action pair can be readily recovered with dynamic programming – transforming the optimization over the entire policy in (5.4) to a step-wise 'min-max' over each control sequence $\{u_t^{p_1}, u_t^{p_2}\}_{0 \le t \le T}$. Under ideal conditions, it is desirable to determine the optimal value function under perfect play, however, given the curse of dimensionality for BOO problems, the best we can hope for is an approximately optimal value $v_\psi^\star(\mathbf{x})$ by continually estimating the value function $v_\psi^p(\mathbf{x})$ using e.g. a policy parameterized by a large function approximator such as deep neural networks $\boldsymbol{f_\psi}$ to approximate the optimal value so that $v_\psi(\mathbf{x}) \approx v_\psi^p(\mathbf{x}) \approx v^\star(\mathbf{x})$. Here ψ are Lipschitz basis functions that are parameters of the function approximator. For a detailed exposition of games, we refer readers to (Roger, 1991) and (Basar and Olsder, 1999).

5.2.1 Sparse Lookout Tree Simulation

We iteratively sample beam angles within the setup by carrying out a lookahead search from the tree's root node at a fixed depth. We restrict samples to 90 discretized beams in $\boldsymbol{\Theta}$. At the first iteration, the subset of beam angles are randomly sampled from $\boldsymbol{\Theta}$. We then progressively add children nodes using the *expand_policy* of Algorithm 1, guided by *probabilities* $p(s, u)$ that are generated by a deep neural network policy $\boldsymbol{f}_{\psi}$, that recursively *expand*s the current node or rolls out simulation from the current node to completion. To prevent "angle collisions", we introduce a minimum pairwise distance, $\bar{d}_i \in \mathbb{R}^+$ between the beamlets in a beam block, defined as $\|\theta_i - \theta_j\| \geq \bar{d}_i, \forall \{j \in m \setminus i\}$, with $\bar{d}_i = 20°$. Repeatedly performing roll-outs, a history of state-action value pairs along the tree's edges is kept; thus biasing action selection toward regions of the state space that produce beams that have not been visited and aiding faster convergence if the same state is encountered more than once.

We compute the mean outcome of every simulation through state s in which action u is selected, $i.e.$ the tree's $Q(s, u)$-value, as $Q(s, u) = \dfrac{1}{\mathbb{N}(s, u)} \sum_{i=1}^{n(s)} \mathbb{I}_i(s, u)\zeta_i$, where $\mathbb{N}(s, u) = \sum_{i=1}^{n(s)} \mathbb{I}_i(s, u)$ is the total number of simulations in which action u was selected in state s, $n(s)$ is the total number of times a game is played through state s, and ζ_i is the outcome of the ith simulation played out from s. Specifically,

$$\mathbb{I}_i(s, u) = \begin{cases} 1, & \text{if } u \text{ was selected on the } i\text{'th policy rollout} \\ 0, & \text{otherwise.} \end{cases} \tag{5.5}$$

Our tree search strategy is an adaptation of the UCT algorithm and is given in Algorithm 1. During simulation, each state and action in the search tree are updated as:

$$n(s_t) \leftarrow n(s_t) + 1; \tag{5.6a}$$

$$N(s_t, u_t) \leftarrow N(s_t, u_t) + 1 \tag{5.6b}$$

$$Q(s_t, u_t) \leftarrow Q(s_t, u_t) + \pm r(s_t, u_t), \tag{5.6c}$$

Algorithm 1 Deep BOO Monte Carlo Tree Search

function **MCTS**($\mathbf{x}_0$)
  $\mathbf{x}_0 \leftarrow \boldsymbol{x}_0(\mathbf{x}_0)$
  while `search_time` $<$ `budget` **do**
    $\bar{\boldsymbol{x}} \leftarrow$ **EXPAND_POLICY**($\boldsymbol{x}_0$)
    $\bar{\boldsymbol{x}}.r \leftarrow$ **FMO_POLICY**
    BACKUP($\bar{\boldsymbol{x}}, \bar{\boldsymbol{x}}.r$)
  end while
  return **BEST_CHILD**($\boldsymbol{x}_0, c$)
end function

function **SELECT_MOVE**($\boldsymbol{x}, c$)
  if p_1 `to play` **then**
    return $\text{argmax}_{\bar{\boldsymbol{x}} \in \boldsymbol{x}} Q(\bar{\boldsymbol{x}}) + \mathcal{K}(\bar{\boldsymbol{x}})$
  else
    return $\arg\min_{\bar{\boldsymbol{x}} \in \boldsymbol{x}} Q(\bar{\boldsymbol{x}}) - \mathcal{K}(\bar{\boldsymbol{x}})$
  end if
end function

function **EXPAND_POLICY**($\boldsymbol{x}$)
  while $\boldsymbol{x}$ `nonterminal` **do**
    if $\boldsymbol{x}$ `not f.expanded` **then**
      return **EXPAND** ($\boldsymbol{x}$)
    else
      $\boldsymbol{x} \leftarrow$ **BEST_CHILD**($\boldsymbol{x}, c$)
    end if
  end while
  return $\boldsymbol{x}$
end function

function **BEST_CHILD**($\boldsymbol{x}_0$)
  if p_1 `to play` **then**
    return $\boldsymbol{x}_0[\arg\min$ `children of` $\boldsymbol{x}_0]$
  else
    return $\boldsymbol{x}_0[\arg\max$ `children of` $\boldsymbol{x}_0]$
  end if
end function

function **FMO_POLICY**($\boldsymbol{x}$)
  return $r = -h^{\star}(\boldsymbol{x}(\mathbf{x})|\cdot)$
end function

function **FULLY_EXPANDED**($\boldsymbol{x}, \boldsymbol{d}$)
  $d_i \leftarrow$ `pairwise_distance`($\boldsymbol{x}.\mathbf{x}$)
  `min_elem` $\leftarrow \min(\boldsymbol{d})$
  if `min_elem` $< \boldsymbol{d}$ **then**
    return `True`
  else
    return `False`
  end if
end function

function **EXPAND**($\boldsymbol{x}$)
  $\bar{u}$ = **SELECT_MOVE**($\boldsymbol{x}$, c)
  `sample` $\bar{\theta}$ `with` $\boldsymbol{x}.p(s, a)$
  update $\bar{\theta} \leftarrow \bar{\theta} + \bar{u}$
  with $\boldsymbol{\pi}_{t-1}$, `create` $\bar{\boldsymbol{x}}.p(\bar{\mathbf{x}}, \bar{u})$
  while `not` $\bar{\boldsymbol{x}} \in \boldsymbol{x}_0$ **do**
    `add` $\bar{\boldsymbol{x}}$ `to` $\boldsymbol{x}$
  end while
  return $\bar{\boldsymbol{x}}$
end function

function **BACK_UP**($\boldsymbol{x}, c$)
  while $\bar{\boldsymbol{x}}$ `not null` **do**
    $N(\bar{\boldsymbol{x}}) \leftarrow \bar{\boldsymbol{x}} + 1$
    $Q(\bar{\boldsymbol{x}}) \leftarrow Q(\bar{\boldsymbol{x}}) + \bar{\boldsymbol{x}}.r$
    $\bar{\boldsymbol{x}} =$ `parent of` $\bar{\boldsymbol{x}}$
  end while
end function

where $\mathcal{K}(\bar{\boldsymbol{x}}) = c\sqrt{\dfrac{2 \ln n(\bar{\boldsymbol{x}}.\mathbf{x})}{N(\bar{\boldsymbol{x}}.\mathbf{x}, u)}}$ and $\bar{\boldsymbol{x}} \in \boldsymbol{x}$ implies $\bar{\boldsymbol{x}} \in$ children of $\boldsymbol{x}$.

where $r(s_t, u_t)$ is the reward/cost gained or incurred by the agent after action u in state s_t. This MCTS scheme makes beam angle transitions evolve in a highly selective, best-first behavior – expanding promising areas of the search space deeper, given infinite memory and computation. After each simulation, a 'best move' for the current beam block is selected; we exponentiate the move probabilities by a temperature slightly larger than unity to encourage diversity in early play; specifically, we compute

$$p(u|s_0;\psi) = \frac{N(s_0,a)^{1/\tau}}{\sum_b N(s,b)^{1/\tau}}, \tag{5.7}$$

where τ is the temperature factor that diversifies the move probabilities. The modified UCT algorithm applied to optimal beam angle selection is presented in algorithm 1.

In order for the state definition to capture as much information as possible under uncertainty, we define an adaptive allocation rule for determining the transition between states since we do not know what node may yield the best bandit: a player might be biased towards always selecting the beams set with the maximum value such that as the search progresses, the worst possible bias is bounded by a quantity that converges to zero. Therefore, we define the state broadly enough to capture all subjective unknowns that might influence the *payoff*/*reward* to be received by a rational decision-making agent; we thus leverage the *upper confidence bound* algorithm of (Agrawal, 1995) to assure an asymptotic logarithmic regret behavior. The state at time step, t, contains a pointer to the patient's CT mask, the number of games that have so far been played through it, and the beam angles contained at that state.

We attach a regret term $U(n(\mathbf{x}))$ to the Q-value so as to ensure the optimal beam does not evade the simulation $i.e.$ $Q(\mathbf{x},u) - U(n(\mathbf{x})) \leq Q(\mathbf{x},u) \leq Q(\mathbf{x},u) + U(n(\mathbf{x}))$; the width of this confidence bound guides the exploration strategy for states that are momentarily unpromising in values but may later emerge as promising states.

Definition 3. We define an *upper confidence bound* $U(s,u)$ on $Q(s,u)$ that adds an exploration bonus that is highest for seldomly visited state-action pairs so that the tree expansion policy selects

the action $u^{\star}$ that maximizes the augmented value:

$$\bar{Q}(s,u) = Q_j(s,u) + c\sqrt{\frac{2\ln\, n(s)}{\mathbb{N}(s,u)}}, \quad \text{where } a^{\star} = \arg\max_{u} \bar{Q}(s,u). \tag{5.8}$$

$Q(s,u)$ is the highest average observed reward from node j – encouraging exploitation of the current node, and ln $n(s)$ is the natural logarithm of the total number of roll-outs through state **x**. The second term in (5.8) encourages exploration of other beam angles, and c is a scalar exploration constant.

We continually update the weights of the neural network policy in a separate thread, writing the weights to a shared memory buffer for the MCTS to read from, $i.e.$ the search thread uses the previous iteration of the trained network policy to to run the policy improvement procedure. When angles are at the edges $i.e.$ 0° or 360° and an angle change outside the range $0 \leq \theta \leq 360$ is recommended, we "wrap" around to enforce cyclicity. Note that the **`EXPAND_POLICY`** and **`FMO_POLICY`** procedures of Algorithm 1 can be seen as a form of Add/Drop simulated annealing as described in (Aleman et al., 2008b). While the **`FMO_POLICY`** procedure returns the node with the optimal fluence cost, the **`BEST_CHILD`** procedure compares the quality of all beam angle sets in the children of the tree's root node.

5.3 Approximate Dynamic Programming

ADP can be considered as a trial-and-error search problem via interaction with an environment (e.g. a dynamical system); the purpose of the interaction is to discover a mapping between events and actions in order to maximize an extrinsic reward signal; this is usually achieved with optimal control of incompletely known MDP. The MDP consists of states $\mathcal{S}$, actions, $\mathcal{A}$, a transition probability, $\mathcal{P}^{a}_{ss'}$ and a reward function $\mathcal{R}^{a}_{s}$. $\mathcal{P}^{a}_{ss'}$ is a probability distribution that governs the evolution of states e.g. from $\mathbf{x} \rightarrow \mathbf{x}'$, while $\mathcal{R}^{a}_{s}$ determines a reward after the transitions occur. In *Q-learning*, an agent's experience is made up of distinct episodes: at the t'th episode, the agent observes its

current state, $\mathbf{x}_t$, selects and executes an action u_t, which transitions the agent to a new state $\mathbf{x}_{t+1}$; an immediate reward, r_t, is subsequently given to the agent, by which the previous Q-values are adjusted. Q-learning relies on *value iteration* to reach convergence. In practice, this takes a long time to complete. *Batch Q-learning* algorithms e.g. (Mnih et al., 2015) attempt to learn a stochastic policy $\boldsymbol{\pi}_\psi : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{R}$ that maximizes the cumulative discounted reward $\sum_{t=0}^{T-1} \gamma^t r(\mathbf{x}_t, u_t)$, where ψ are the parameters of a policy.

5.3.1 Self-Play Neuro-Dynamic Programming

We consider an ADP setting whose task is to discover the mapping between patient's geometry and a good beam angle combinations by maximizing an extrinsic reward signal that is stage-wise informed by the quality of a fluence profile. To this end, we apply optimal control of incompletely known *MDP*s (Astrom, 1965). The MDP consists of states $s \in \mathcal{S}$, actions, $a \in \mathcal{A}$, transition probability, $\mathcal{P}^a_{ss'}$, and a reward function $\mathcal{R}^a_s$. $\mathcal{P}^a_{ss'}$ governs states evolution from $s \rightarrow s'$, while $\mathcal{R}^a_s$ determines stage-wise rewards after transitions.

To find a good saddle-point for the optimization problem, we applied weakened FSP (Heinrich et al., 2015) to find an *approximate* best response strategy to an opposing agent's mixed strategy in Markov decision games of self-play. In our formulation, each player does not know the strategy of its opponent ahead of time, $i.e.$, their security levels do not necessarily coincide. To ensure that equilibrium can be found within *pure strategies*, we let one player act after observing the decision outcome of the other player.

The network, $\boldsymbol{f}_{\boldsymbol{\psi}}$, predicts a probability distribution over all beam angle configurations, $p_a = p(s, \boldsymbol{u})$, and a *value*, $v_\psi(\mathbf{x})$ – an estimate of the optimal beam angle set θ is the optimal beam set. For a game Γ, suppose that $y = \{y_1, \ldots, y_m | \sum_{i=1}^m y_i = 1\}$ and $z = \{z_1, \ldots, z_n | \sum_{i=1}^n z_i = 1\}$ are the respective probability distributions for players p_1 and p_2. The average value of the game will correspond to player p_1 minimizing a cost $\mathcal{J}(y, z) = y^T \Gamma z$ and player p_2 maximizing $\mathcal{J}(y, z)$. Each player's action is governed by a mixed strategy – obtained by adding a Gaussian random

walk sequence with standard deviation 2 to the prior probability distribution predicted by the neural network policy or computed by the tree policy; this is then normalized by the sum of the resulting noised distribution. Players $p_1,$ and p_2's strategies are independent random variables, repeatedly implemented during game simulations. As the number of times the game is played gets larger, the frequency with which different actions for p_1 and p_2 are chosen will converge to the probability distribution that characterize their random strategies (Basar and Olsder, 1999, pp.24).

Each MDP episodic setting involves randomly sampling from the CT dataset, concatenating the sampled geometry to the previous beam blocks, and eliminating the beam that has been longest in the block as in a *FIFO* scheme. States are modified based on the probability predicted by policy, $\boldsymbol{\pi}(\cdot)$, which along with the patient's state, $\mathbf{x}_t$, (including the current and previous five beam block configuration) result in a new node. Between node transitions, a full FMO is carried out – evaluating the quality of chosen beamlets, and informing updates in search probabilities that we later compute when we reach a terminal leaf.

The network policy, $\pi(\cdot|\psi_t)$, and search tree, $\Gamma(\boldsymbol{\pi}_\psi(\cdot))$, are optimized in separate concurrent threads; to assure non-blocking of search and network optimization processes, the network's weights were written to a shared memory map, where they are asynchronously updated by gradient descent, while the tree search thread ran in a parallel thread from a previous iteration of the network policy, $\pi(\cdot|\psi_{t-1})$. At the end of each MDP and MCTS simulation, we compare the value predicted by either player, average their mixing strategies and update the gradients of the loss function with respect to the *values* based on equation (5.9).

We train the probability distribution over current beams by maximizing the similarity between the computed search probabilities $\boldsymbol{\pi}$ and the predicted distribution $\boldsymbol{p}$ (by the search process) with the cross-entropy loss:

$$\Delta\psi_p = \frac{\log \partial \boldsymbol{p}_\psi(\boldsymbol{u}|\mathbf{x})}{\partial\psi}(\boldsymbol{\pi}^T\,\mathbf{p}),$$

and we take the network weights' induced tensor norm (last term of (5.9) given its robustness to the asymmetrical network modular weights). Altogether, we minimize the combined loss,

$$l = (\boldsymbol{\zeta} - \boldsymbol{v})^2 - \boldsymbol{\pi}^T \mathbf{log}(\boldsymbol{p}) + \lambda \|\boldsymbol{\psi}\|_2^2, \tag{5.9}$$

where λ (set to 1.5) controls regularization of the network weights to avoid overfitting. The cross entropy term was weighted by 0.9, and the mean square error (mse) loss by 0.01 to keep the overall loss in the direction of persistent reduction in training error. These values were found by empirical evaluations of the loss surface. We use 4 search threads, 8 CPUs, and 4 GPUs for the final version of this algorithm. At each terminal node of the tree, new search probabilities are computed, exponentiated by the inverse of a temperature parameter (set to 0.98).

5.4 Results

This section presents results on 3D prostate cases. We start the training process by randomly adding five beam blocks at first to the state queue. The input planes are then passed through the tower residual network, from which probability distributions and a value are predicted. We add a random walk sequence to the generated pure strategy from the neural network and construct our tree with this mixed strategy (at subsequent iterations, a previous iteration of the neural network is used to generate the mixed strategy used to run the tree search). As new beam angle combinations are found according to Algorithm 1, the FIFO queue is updated accordingly. At the end of every tree search, we compute new search probabilities and compute the backward pass of the algorithm. This is where the weights of the network are updated based on the loss function proposed in equation (5.9).

The mixed strategy is a by-product of the prediction by the deep neural network policy; it guides search for optimistic beam angles, based on a reward (or cost) that is maximized (or minimized) by either of players p_1 or p_2. When a move is selected (see Algorithm 1), the current node is expanded, resulting in a new set of beamlets. The fluence (using eq. (5.1)) for the beamlets are optimized, and

the reward for maximizing player, or the cost of the minimizing player are updated. We continue the expansion of the leaves until we reach a terminal leaf.

The displayed plots in Figure 5.5 are the dose washes we obtain by optimizing the fluence (a resultant of the beam angles chosen) on the patient. The wash plots are effects achieved by the display of the voxel intensity distributions (in 2D) for organs and tumors present in a target volume slice. They enable the interpretation of the quality of a dose based on the organs that must be targeted ($i.e.$ tumors) and organs that must be avoided ($i.e.$ critical structures). The dosage is represented as a heat map. Regions that receive zero radiation are dark blue, while regions that receive higher doses progress in the order of higher intensity from green through yellow to red. As seen, the intersection of beams delivers heavy doses to the tumors while largely sparing moderate surrounding tissues. The line overlays on the plots depict the angle of incident radiation. The policy selects fairly equidistant beams, yielding wash plots that provide good dosimetric concentration on the tumor (center of the slices shown) and sharp gradients at the transition between tumors and OARs, while also largely avoiding strong dose to OARs.

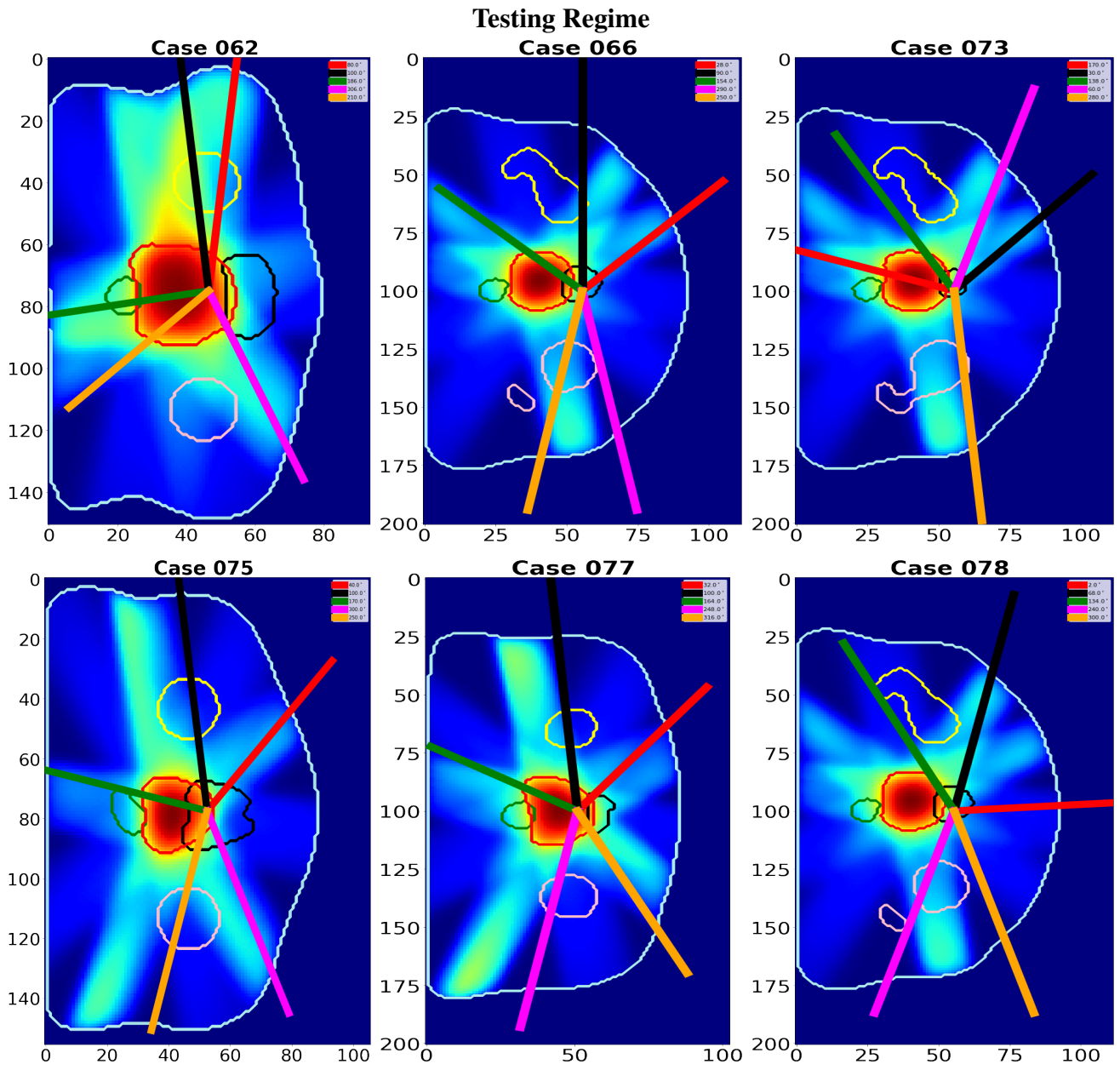


Figure 5.5: Dose distribution for select patients during testing of self-play network

CHAPTER 6

IMPROVING BEAMS SELECTION WITH SUPERVISED PRE-TRAINING

In this chapter, we improve upon the results presented in Chapter 5 by pre-training a deep neural network Algorithm in a supervised setting. The pre-trained network will then be used to hasten beams selection/prediction on anatomical structures of a patient's body on test cases. To allow generalization to future volumetric arc radiation therapy settings, we will conduct the tree search in such a way that the nodes traversal during search can generate as many beam plans as a physician would want. We introduce a search strategy that initially samples from the global beam angle space given a generative Markov decision process (MDP) policy (henceforth referred to as a base policy – in this case a deep policy that parameterizes a column-generation type linear program that learned the mapping from patient CT maps to beam orientations). This is then followed by a stochastic beams selection strategy that identifies promising beam angle candidates using a sparse lookahead tree search.

6.1 Generative MDP Prior

As alluded to earlier, the BOO problem is mathematically intractable lacking a proper mathematical system model. As such, it is difficult to model the transition probabilities and system equations. Whereas in (Ogunmolu et al., 2018), we implemented a tower residual deep neural network (DNN) that acts as a model simulator for the transitions between states, in (Sadeghnejad Barkousaraie et al., 2019), a more adaptive prior was developed that iteratively predicts beamlets to be added to a current beam plan using a column generation procedure. Given a set of dose data, structure weights and selected beams for a set of patients, we construct a column-generation (CG) linear program whose goal is to build a mapping, Θ, between patients' anatomy and beam angles. Here, we choose a more compact representation of the beam angles. Specifically, we stack the organs in a patient's CT scan together in a 2D map of slices; these are then joined with a Boolean array that represents the orientation indices in consideration at time k as shown in Figure 6.1.

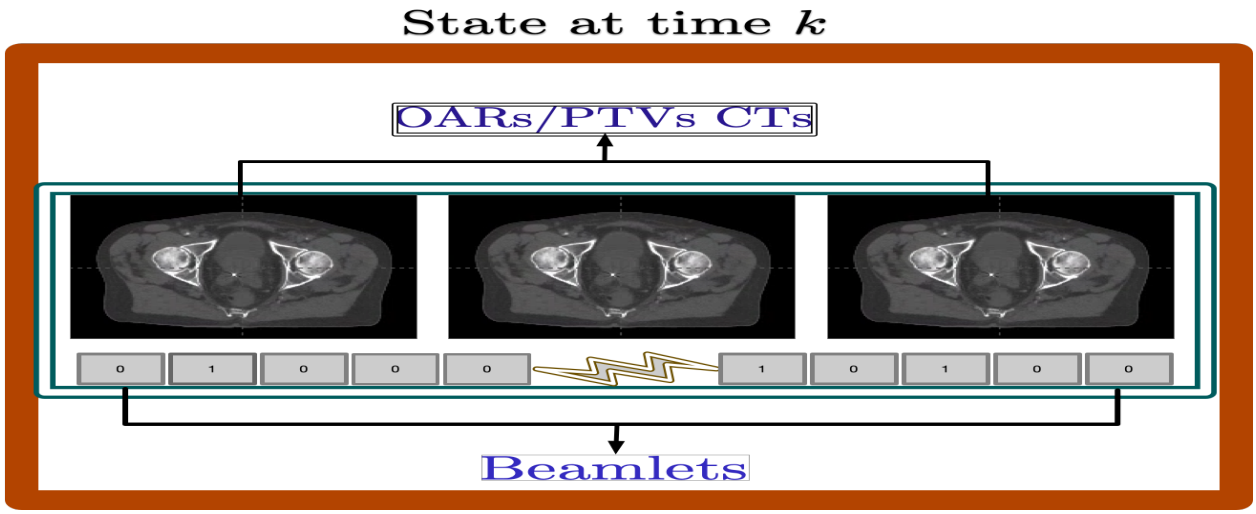


Figure 6.1: System state $\mathbf{x}_k$ consists of the patient's CT mask as well as a binary valued row vector that represents beamlets under consideration at time k. The indices with 1 denote beams that are currently turned on for this patient.

Column generation, in external beam radiation therapy, has proven effective in previous beam orientation optimization problems (Long, 2015; Romeijn et al., 2005). Since our goal is to mitigate online replanning when a new patient comes to the clinic, we again use precomputed dose influence matrices, $\mathcal{D}_{ij}$, for previously treated patients, as well as beam weights necessary for FMO during search. Consider the FMO problem description of § 5.1.4. For a subset, B_{lim}, of the available discretized beams set, B, the optimal beams set $B^{\star}_{\text{lim}}$ are found via greedy linear programming strategy – essentially, a column generation (CG) procedure. We iteratively add beams to the current set of existing beams B_{lim} until the cardinality of B_{lim}, $|B_{\text{lim}}|$, reaches an upper threshold, $\bar{B}_{\text{lim}}$ or when an optimal condition for the master problem (optimizing for the beams in the universal beams set) is achieved. A beam is added only if it minimizes the current objective function value. We reprint the the Algorithm from (Sadeghnejad Barkousaraie et al., 2019a) in 2.

6.1.1 Base Policy Approximation

A deep policy parameterizes the column generation functional mapping from patient geometries to beam angles. The search proposal presented in the rest of this chapter can use any other appropriate

Algorithm 2 Column Generation Procedure

Require:

$\bar{B}_{\text{lim}}$ ▷ upperbound on $|B_{\texttt{lim}}|$

Initialize $B_{\texttt{lim}} = \{\}$ ▷ an empty set

Initialize $k = 0$ ▷ the number of iterations

while $|B_{\text{lim}}| < \bar{B}_{\text{lim}}$: **do**

  $b^{k+1} = \text{argmin}_{u \in B_{\text{tot}} \setminus B^k_{\text{lim}}} \{P(\hat{B}) : \hat{B} = B^k_{\text{lim}} + u, |\hat{B}| = |B^k_{\text{lim}}| + 1\}$

  $B^{k+1}_{\text{lim}} = B^k_{\text{lim}} + b^{k+1}$

end while

base generative model of an MDP that is available. We will refer to the deep policy as the base policy henceforward. The base policy has an anatomical input and a beam orientation input. The anatomical input has two channels, each of size $N \times 64 \times 64$. The first channel consists of PTVs while the second channel encodes the OARs as an avoidance map. This allows separating the PTV from the OARs. The beam orientation input is a zero-initialized vector whose particular indices are complemented over the course of backpropagation. The approximated policy was earlier trained to learn network weights that minimize the difference between a column generation based prediction $\mathbf{x}^{cg}_{k+1}$ and targets $\mathbf{x}^{kkt}_{k+1}$ from the Karush-Kuhn-Tucker (KKT) conditions (Karush, 1939; Kuhn and Tucker, 1951). The base policy outputs a probability distribution whose samples inform the beam index to which a new beamlet $i.e.$ $\mathbf{x}_{k+1}$ is inserted according to Figure 6.1. This probability distribution is a rational decision-making agent's preference for a *lottery* in an uncertain environment. It characterizes a policy's belief about all relevant unknown factors at the current time step k given a decision u_k. As new information becomes available, the decision-maker's subjective probability distribution gets revised. Repeatedly sampling from this probability distribution enables the transition between episode contexts, $i.e.$ $\mathbf{x}_k \rightarrow \mathbf{x}_{k+1}$. Further details about this network policy are fully described in (Barkousaraie et al., 2019) and (Sadeghnejad Barkousaraie et al., 2019a). The remaining contents of this chapter relates to our tree search contribution.

6.2 Sparse Lookahead Search

Suppose we have a base policy $\{\mu_0, \ldots, \mu_{N-1}\}$, over an horizon N, which serves as a generative MDP policy. We can determine a rollout control for each state $\mathbf{x}_k$ at time k by using controls $u_k \in U_k(\mathbf{x}_k)$ to produce a large number of simulated beam plans starting from an initial state-control pair, $(\mathbf{x}_k, u_k)$. Simulated beam plans may be a consequence of

$$\mathbf{x}_{i+1} = f_i(\mathbf{x}_i, \mu_i(\mathbf{x}_i), w_i), \quad i = k+1, \ldots, N-1 \tag{6.1}$$

where $\{\mu_{k+1}, \ldots, \mu_{N-1}\}$ represents the tail part of the base policy so that the first generated state is of the form

$$\mathbf{x}_{k+1} = f_k(\mathbf{x}_k, u_k, w_k) \tag{6.2}$$

with independent random samples $\{w_k, \ldots, w_{N-1}\}$. A beam angle combination search task in this setting can be defined by a cost function $Q(\mathbf{x}_k, \boldsymbol{u}_k)$, recoverable from a policy $p(u_k|\mathbf{x}_k; \mu)$, that is distributed over controls, u_k, conditioned on the states, $\mathbf{x}_k$, and parameterized by a rollout (tree) policy, μ. For clarity, we will denote the control conditional $p(u_k|\mathbf{x}_k, \mu)$ as $\boldsymbol{\pi}_\mu(\boldsymbol{u}_k|\mathbf{x}_k)$.

Suppose we simulate the *cost-to-go* approximation for the unknown transition probabilities and system equations (6.1) by using in part Monte-Carlo simulations. The state-action costs of running many trajectories of the form (6.2) are considered as samples from an expectation of Q, $i.e.$,

$$Q_k(\mathbf{x}_k, u_k) = \mathbb{E}\{g_k(\mathbf{x}_k, u_k, w_k) + J_{k+1}(\mathbf{x}_k + 1)\} \tag{6.3}$$

where $J_{k+1}(\mathbf{x}_k + 1)$ denotes the cost of running the base policy starting from $\mathbf{x}_{k+1}$, and $g_k(\mathbf{x}_k, u_k, w_k)$ is the cost of running the current tree policy. We average the costs $Q_k(\mathbf{x}_k, u_k)$ of each control $u_k \in U_k(\mathbf{x}_k)$ to obtain an estimated cost $\bar{Q}_k(\mathbf{x}_k, u_k)$. The approximate rollout control $\bar{\mu}_k(\mathbf{x}_k)$ can be computed with the minimization

$$\bar{\mu}_k(\mathbf{x}_k) = \arg \min_{u_k \in U_k(\mathbf{x}_k)} \bar{Q}_k(\mathbf{x}_k, u_k). \tag{6.4}$$

Equations (6.1), (6.2), and (6.3), are still unrealistic for our BOO problem in terms of computational economy. This is due to the length of the horizon N, which may elongate the trajectories when N is large, resulting in controls u_k that may be inferior – not meriting much sampling effort, or controls u_k that appear temporarily promising but may become inferior as other parts of the state space become explored (Bertsekas et al., 2017a). Therefore, we again introduce a variant of the Monte-Carlo Tree Search (MCTS) paradigm (Browne et al., 2012; Coulom, 2006; Gelly and Silver, 2011; Kocsis and Szepesvári, 2006; Ogunmolu et al., 2018) to mitigate against these possible drawbacks. MCTS iteratively runs random simulations from a tree's root position through its nodes by randomly choosing actions until arrival at a terminal state. A back-up operator progressively averages the outcome of many random simulations to min-max as the amount of simulation expands. For a state $\mathbf{x}_t$ other than the root node $\mathbf{x}_0$ in a tree, a *tree policy* decides whether to recursively *expand* the current state by executing controls $\{u_0, u_1, \ldots, u_k\}$ at every time step, k, or if it should roll out the current game to completion by finding the best child of the root node. During each expansion process, a child node is only added to the tree if it is not already part of the tree, and the child node's reward is computed; this saves memory and computational requirement. Essentially, games of self-play are simulated from the root node, $\mathbf{x}_0$ of a tree, and the Monte Carlo value, $Q(\mathbf{x}, u)$, of the tree ($i.e.$ the mean outcome of every simulation through state $\mathbf{x}$ in which action u was selected) is computed as in (5.5). During simulations, each state and action in the search tree is updated as in (5.6)

6.2.1 Game Tree Simulation

We run a game simulation of *perfect recall*, where each node's current *information state* implies knowledge of the ancestral node sequences that led to it. We use a variant of MCTS to average our Monte-Carlo evaluations, based on a sparse look-ahead search strategy for large state-space MDPs. Single agent games are simulated from a tree's root node $\mathbf{x}_k$ at time k up to a fixed depth using a child selection policy drawn from the base policy $\{\mu_0, \ldots, \mu_{N-1}\}$. , Figure 6.2 illustrates

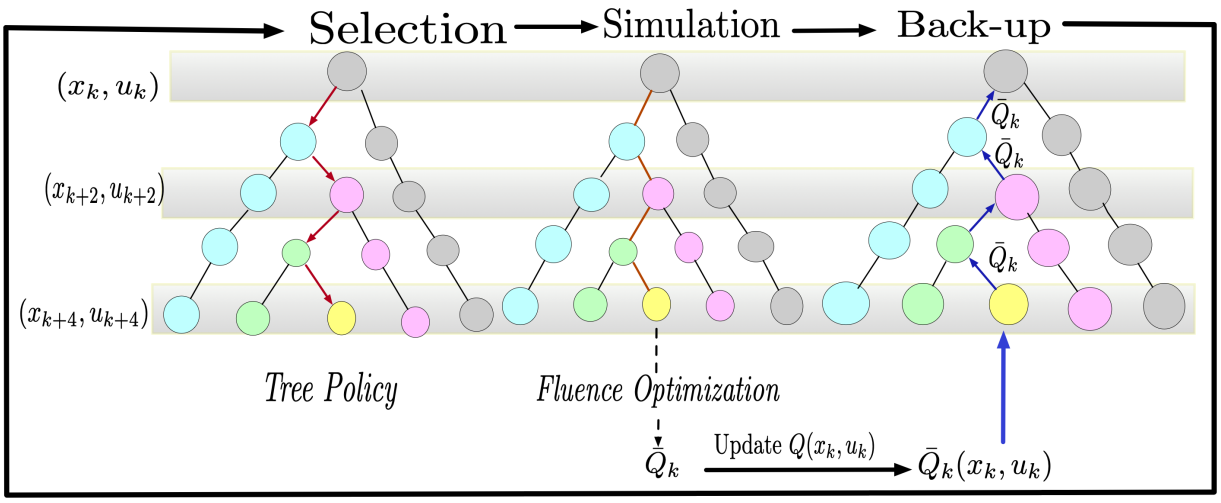


Figure 6.2: A single iteration of our rollout lookahead tree search.

the search transitions up to a fixed depth. Here, each node contains state-control pairs $(\mathbf{x}_k, u_k)$, connected by edges which are labeled by their state-action values, $\bar{Q}_k$. Equally colored beamlets between episodic transitions, $i.e.$ $\mathbf{x}_i \rightarrow \mathbf{x}_{i+1}$, indicate beamlets being passed onto the following episode. The red arrows indicate the path through which new beamlets are added. Once the number of beam plans are reached, we finish the game, find the optimal FMO of the terminal leaf, and approximate leaves beneath this leaf by this *value*. The result of the MC evaluation is backed up to the root node. A partial tree of beam angle candidates is randomly constructed, starting from a root node, and we progressively add child nodes to the current node in a ***selection*** process guided by the base policy, $\{\mu_0, \ldots, \mu_{N-1}\}$. Popular MCTS implementations use a rapid rollout of multiple Monte-Carlo simulations to traverse from root node through a leaf node while adding children nodes to the tree at each successive rollout episode. At the leaf node, results from multiple games are averaged using a min-max evaluation that decides which of MDP agent won a game. These values are generally drawn from the set $\{-1, 0, 1\}$, corresponding to loss, draw or a win by the current agent so that the average score of is in the interval $[-1, 1]$.

While the upper confidence bound applied to trees (UCT) Algorithm (Kocsis and Szepesvári, 2006) improves the performance of rollout-based Monte Carlo planning strategies for Go games

(Gelly and Silver, 2011; Silver et al., 2017), applying it to the BOO problem requires rethinking the MCTS' self-play strategy. We are interested in a formulation where a single agent performs the tree search. Single-player Monte-Carlo tree search (SPMCTS) (Schadd et al., 2008) is a proposed modification to the UCT Algorithm of (Kocsis and Szepesvári, 2006) that resolves the exploration-exploitation tradeoff in single-agent systems. The effectiveness of SPMCTS was demonstrated when it achieved the highest score on the "SameGame" puzzle by balancing the exploitation-exploration tradeoffs under long time controls as well as by combining a large number of small searches instead of rolling out a large search.

We formulate our rollout control search as a bandit problem, where internal nodes are considered as K-arms. Since $\bar{Q}$ is estimated for the beam angles explored in the past, it is difficult to solely base decisions for optimal beam based on past cost distributions alone: one must balance the exploitation of the control currently found to be optimal with the exploration of other control sequences that appear sub-optimal but may become better in performance as we explore other parts of the state space.

We want an adaptive allocation rule for determining the transition between states such that as the search process progresses and achieves convergence as the simulation grows, the worst possible bias is bounded by a quantity that converges to zero. Given the uncertainty of not knowing what node that may yield the best bandit, an agent may be biased towards always selecting beams set with the minimum FMO cost. Typical tree search policies minimize the cost of the form in (6.3) in conjunction with a bias sequence that determines a game's loss based on the current expected cost and the loss for not playing the best bandit. A typical approach is to rewrite (6.3) as

$$\bar{\mu}_k^{\star}(\mathbf{x}_k) = \bar{\mu}_k(\mathbf{x}_k)n \pm \mu_j \sum_{j=1}^{K} \mathbb{E}[T_j(n)] \tag{6.5}$$

where $\mathbb{E}[T_j(n)]$ denotes the expected number of plays for node j over n trials – essentially a regret term, cost $Q(\mathbf{x}, \boldsymbol{u})$ that ensures the optimal beams combination does not evade the simulation. The width of the confidence bound in equation (6.5) guides the exploration strategy for states that are

momentarily unpromising in values but may later emerge promising as the number of games evolve in length.

Definition 4. To address the single-agent search, we propose the following modification to the UCT Algorithm

$$\bar{Q}(\mathbf{x}_k, \boldsymbol{u}_k) = Q_j(\mathbf{x}_k, \boldsymbol{u}_k) + c_p \cdot \pi(\mathbf{x}_k)\sqrt{\frac{ln\, N_p}{1+N_c}} + \kappa \tag{6.6}$$

where N_p denotes the number of times a parent node p was visited, N_c denotes the number of times a game was played through parent p's child c, and c_p and κ are constants that bias search towards rarely visited nodes. Here, $\bar{\mu}_k$ is the average value of a leaf node $i.e.$ the optimal FMO cost, and the penultimate and last terms encourage exploration of other components of the tree. We set c_p to 96 and κ to 10.

Our tree search framework consists of the following stages:

Selection: Children are added to the root node by randomly sampling $\mathbf{x}_{k+1}$ from the base policy, $\{\mu_0, \ldots, \mu_{N-1}\}$. The leftmost tree of Figure 6.2 illustrates how the tree is constructed in a single iteration: red arrows depict the selection strategy traversing the tree from the root to a leaf. Note that the color of every beamlet per episode is unique. This is locally enforced during the tree construction so as to ensure a good coverage of beams in the plan that we settle upon at the leaf node.

Simulation: By continually sampling episodes from the initial state, we keep score of the action values at states sampled in earlier episodes such that if a control is encountered again during search, we can bias future controls toward regions of the state space that yield a favorable outcome. Equation (6.6) guides the tree toward a *best-first* set of beam angle candidates. The best-first leaf node encountered is given by the child node with the lowest cost in the tree.

Expansion: After each simulation, we store the all the children for a simulation. This differentiates our approach from other approaches that expanded only one child in the tree. We store all

children encountered up to the root node to accommodate the manner in which the base policy was evaluated. This symmetry of storing the tree policy enables the improvement of the base policy when it is re-optimized based on the tree search-generated data.

Back-up: After expansion, a full fluence map optimization is computed for the beams at the terminal leaf node. The optimal cost of the fluence is computed. And this value is updated for nodes along the edges that led to this node. Note that the value is averaged per visit count for every node in the tree.

6.3 BOO Rollout Search Algorithm

The rollout Algorithm is summarized in Algorithm 3. An agent is constructed, who carries out an extensive rollout search. In the function `search`, while we are within the time allocated for the search, we continually select a child using the child selection policy from the column generation network described earlier. During each iteration, search is conducted until a given fixed depth is reached (this fixed depth corresponds to the candidate number of beam plans). Following this, we replay all controls sampled so far in a simulation and expansion process. When finished, the node statistics are updated in the scenario presented in Figure 6.2.

In the Algorithms, **`procedure`**.**`function`**(`parameter`) calls a procedure's subroutine, **`function`** that takes the entry`parameter`. The **Require** parameters at the head of a **`procedure`** are global parameters accessible by any member function of the procedure or class. The symbol Υ is a buffer used to store the geometry of the patient that is explored as well as the beam angles considered up until depth $m = 5$ as illustrated in Figure 6.1. In Algorithm 0, Γ denotes the computed policy for the tree after a game is rolled out to completion. During simulations, each state and control in the search tree are updated as in $(5.6)|_{a,b}$.

In Algorithm 4, line 21 signifies that new beamlets are added to the current state (illustrated at the bottom of Figure 6.1) when a suitable control that has not been part of the children of the tree is found. This is done to avoid a repetition of angles in the beam plan. As we recursively traverse

Algorithm 3 BOO Tree Search

Require:

a tree height, h,
game length, g,
a computational budget, T
procedure `BOO_Search()`
    function `run_search(base_policy)`
        `agent =` **`search`**`(base_policy)`
        return `agent.search_data()`
    end function
    function **`search`**`(base_policy)`
        `agent` $\leftarrow$ **`AGENT`**`(base_policy)`
        $\boldsymbol{x} \leftarrow$ `agent.root.choose_leaf()`
        $\mathbf{x}_{k+1}$ = `base_policy(`$\boldsymbol{x}.\mathbf{x}_k$`)`
        $\boldsymbol{x}$`.update_stats(`$\boldsymbol{x}, \mathbf{x}_{k+1}$`)`
        while T `> 0` **do**
            `current_`h = $\boldsymbol{x}$`.N()`
            while $\boldsymbol{x}$`.root.N() <` `current_`h `+` h **do**
                `leaves = agent.tree_search(`h`)`
            end while
            $\boldsymbol{x}$`.state.play_controls(controls)`
            if `state.n >` g **then**
                `break`
            end if
            $T = T - 1$
        end while
        return `agent`
    end function
end procedure

the edges of the tree, it is natural that "angle collisions" may occur, $i.e.$ we end up with a node that has more than one or two similar angles, or very close angles in the configuration space of the setup. To avoid this, we re-introduce the minimum pairwise distance, $\bar{d}_i \in \mathbb{R}^+$ between beamlets, originally proposed in (Ogunmolu et al., 2018), and now defined as $\|\theta_i - \theta_j\| \geq \bar{d}_i, \forall \{j \in m \setminus i\}$, with $\bar{d}_i = 15^\circ$, where θ_i's are the beam angles available at a node.

Algorithm 4 Deep BOO 2.0: BOO State

Require:

```
m    ▷ required number of beams in TP
n = 0    ▷ simulation counter
patient  ▷ a patient's CT map
beams    ▷ beams at the current state
procedure State(patient, beams)
    State.x_k = {patient, beams} ▷ patient's CT and current beams (see Figure 6.1)
    function play_controls(u_k, mutate=False)
        if mutate then
            new State.x_k
        end if
        expand(x_k, u_k)
        n = n + 1
    end function
    function optimize_fluence()
        return FMO(State.x_k)
    end function
    function expand(x_k, u_k)
        if x_k + u_k(x_k) = or x_k.size=m then
            Return x_k
        end if
        x_{k+1} ← x_k + u_k(x_k))
    end function
    function end_game()
        return ((n % m) = 0 and (n ≠ 0))
    end function
end procedure
```

6.4 Results

What constitutes an *optimal treatment plan* lacks consensus among radiation oncologists given the non-convexity of the BOO problem. A guiding definition we have used to guide our problem formulation is that the optimal treatment plan is the one which renders the maximal necroticism to tumors whilst sparing healthy tissues and OARs. We have discretized the coplanar geometry of our setup by 2° angular separation – resulting in 180 candidate beams. If the treatment plan were for a non-coplanar plan with a 6° separation between beams, for example, we would have 1162

Algorithm 5 Deep BOO 2.0 Agent

Require:

state obtained from prior policy, $\mathbf{x}_k$
sampled patient CT from CTs dataset
number of branches to add to tree, b
$\boldsymbol{x}^0$ = **NODE**($\mathbf{x}_k$) ▷ root node
procedure AGENT(policy)
    AGENT.policy = policy ▷ base policy
    function Tree_Buffers()
        Initialize buffers X, U, Γ
            ▷ *i.e.* state, controls and tree policy
    end function
    function tree_search(h)
        current_$h = b$
        while current_$h < h$ **do**
            current_$h \leftarrow$current_$h + 1$
            η = **AGENT.**$\boldsymbol{x}^0$**.choose_leaf**()
            cost = $\boldsymbol{x}$**.state.optimize_fluence**()
            $\boldsymbol{x}$**.backup**(**AGENT.**root, cost)
            if $\{\boldsymbol{x}^0$**.Child_**$Q_i \neq 0 \, \forall \, i \in \boldsymbol{x}^0$**.**d$\}$ **then**
                update_buffers() ▷ store MDPs
            end if
            if $\boldsymbol{x}.\mathbf{x}_k$**.end_game() then**
                leaf**.reset_state**() ▷ start along
                    a different branch
            end if
        end while
        if $X \neq \emptyset$ **then**
            $\mathbf{x}_{k+1}^n$ = policy.**predict**(leaves)
            X.**update_stats**($\boldsymbol{x}^0$, $\mathbf{x}_{k+1}^n$)
        end if
        return Υ
    end function
    Continued on next page.

candidate beams. This means there exists a large solution population for the possible beams that can be selected for an IMRT treatment plan. This makes finding *optimal beams* impossible in real or near real-time. Therefore, we choose a base policy, $\{\mu_0, \ldots, \mu_{N-1}\}$, optimized on a greedy CG algorithm to learn the solution space. CG learns a limited solution space by dividing the problem of minimizing a global objective function cost with respect to all available beams into subproblems of

```
    function play_controls(controls)
        X ←insert(x.x_k)
        U ←insert(u_k)
        Γ←insert(x^0.state_control_cost())
    end function
    function search_data()
        X = {x_k, ..., x_{k+m}}^T      ▷ CT masks
        U = {u_k, ..., u_{k+m}}^T      ▷ beam indices
        Γ = {μ̄_k, ..., μ̄_{k+m}}^T      ▷ tree policy
        return X, U, Γ
    end function
end procedure
```

minimizing the objective function with respect to a subspace of the global beams. This subspace is greedily found in terms of the beam areas with a high potential for an optimal treatment plan. We have used a fast first-order primal-dual optimization algorithm (Chambolle and Pock, 2011) to solve the limited problem. Although the Chambolle-Pock algorithm solves an approximated problem, the quality of its solution is high (Sadeghnejad Barkousaraie et al., 2019a). We thus leverage the advantage of CG algorithms (Dong et al., 2013; Woods et al., 2016) and (Dong et al., 2013) in warm-starting the search procedure.

We played single agent games for a total of 100 epochs. Each epoch consisted of multiple simulation roll-outs of beam angles from a tree's root node, guided by a so-called *base policy* from the CG algorithm. Once the number of required beam plans are reached, we restart the simulation, sampling angles for the next iteration with $\{\mu_0, \ldots, \mu_{N-1}\}$ acting on the previously obtained state. The process repeats until a certain amount of branches are added to the tree (see Algorithm 0). To ensure a good coverage of all patients in our data-set, at the leaf node of every episodic setting, we restart the simulation with a new patient's geometry. This is reasonable since we expect that for most prostate cancers, the spatial locations of the organs and PTVs in the human body will be correlated. The tree search endured for 37 hours and 23 seconds on a 500-branch game whose simulation length was 80. What follows in Figures 6.4 and 6.3 are the DVH curves and dose wash plots on the majority of our prostate test cases.

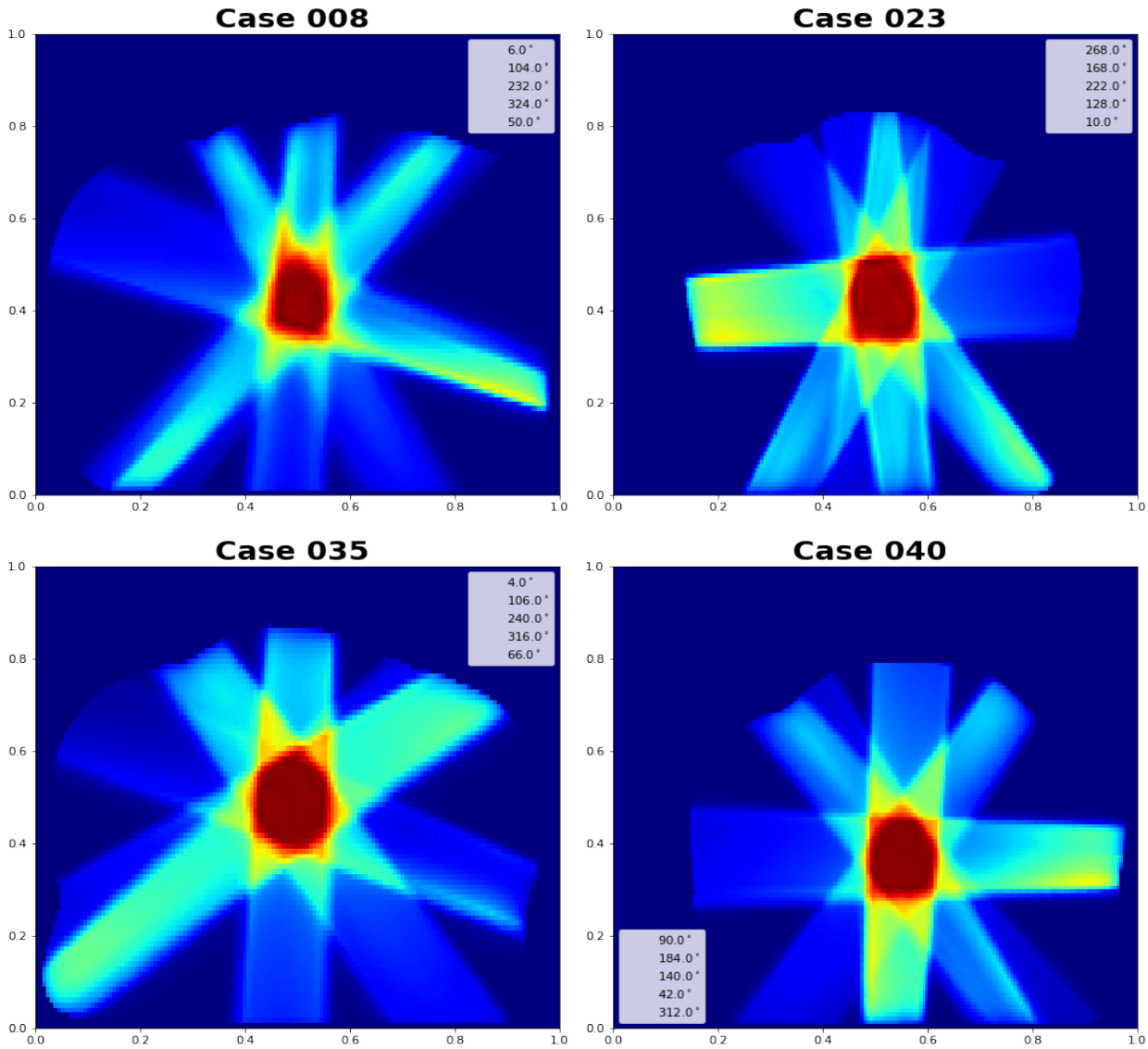


Figure 6.3: Dose distribution of our policy on select prostate cases.

After the tree search procedure is terminated, we collect the tree's data, and project the data back into the cost space of the original base policy. This is consistent with approximate policy iteration schemes (see (Bertsekas, 2013) for an extensive review). The policy $\{\mu_0, \ldots, \mu_{N-1}\}$ is then re-optimized based on the tree data. This yields a policy that is more refined compared against the CG networks of (Barkousaraie et al., 2019; Sadeghnejad Barkousaraie et al., 2019) and (Sadeghnejad Barkousaraie et al., 2019a). We may consider the original policy of (Barkousaraie

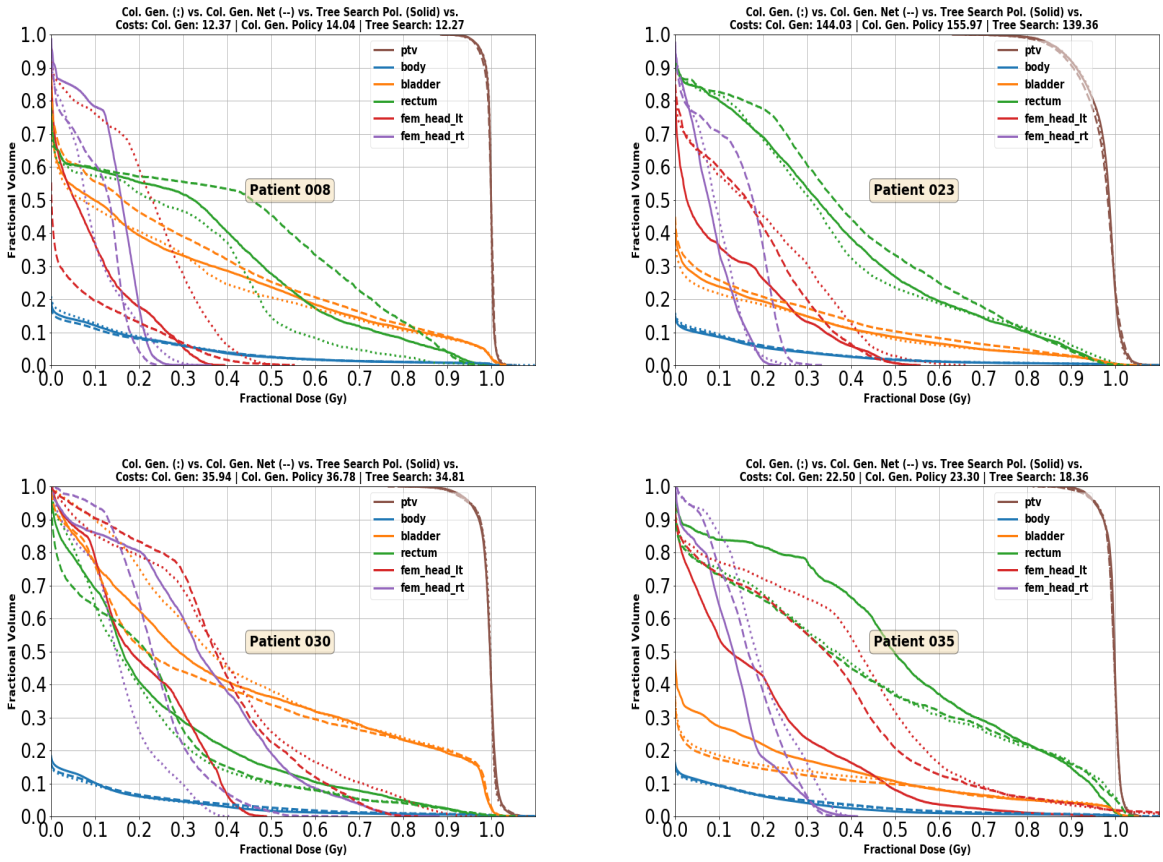


Figure 6.4: DVH curves of our policy versus other schemes.

et al., 2019; Sadeghnejad Barkousaraie et al., 2019) as a policy evaluation scheme, and our proposal as a policy improvement scheme, whereby the initial policy is improved with a more refined data that is projected to its cost function space. After retraining, the network is then used to predict beam plans on test patient geometries. So, suppose a doctor calls for N beam plans, we continually probe the trained neural network on a combination of the patient geometry and previously chosen beams in order to predict the next beam that yields a lower FMO score. Altogether, predicting five beams takes less than 2 minutes.

The Dose-Volume Histogram (DVH) charts of Figures 6.4 and 6.5 are as a result of solving the FMO for the angles predicted by the re-trained neural network on the data gathered from the tree search. The charts compare the performance of our algorithm against plain CG and the network

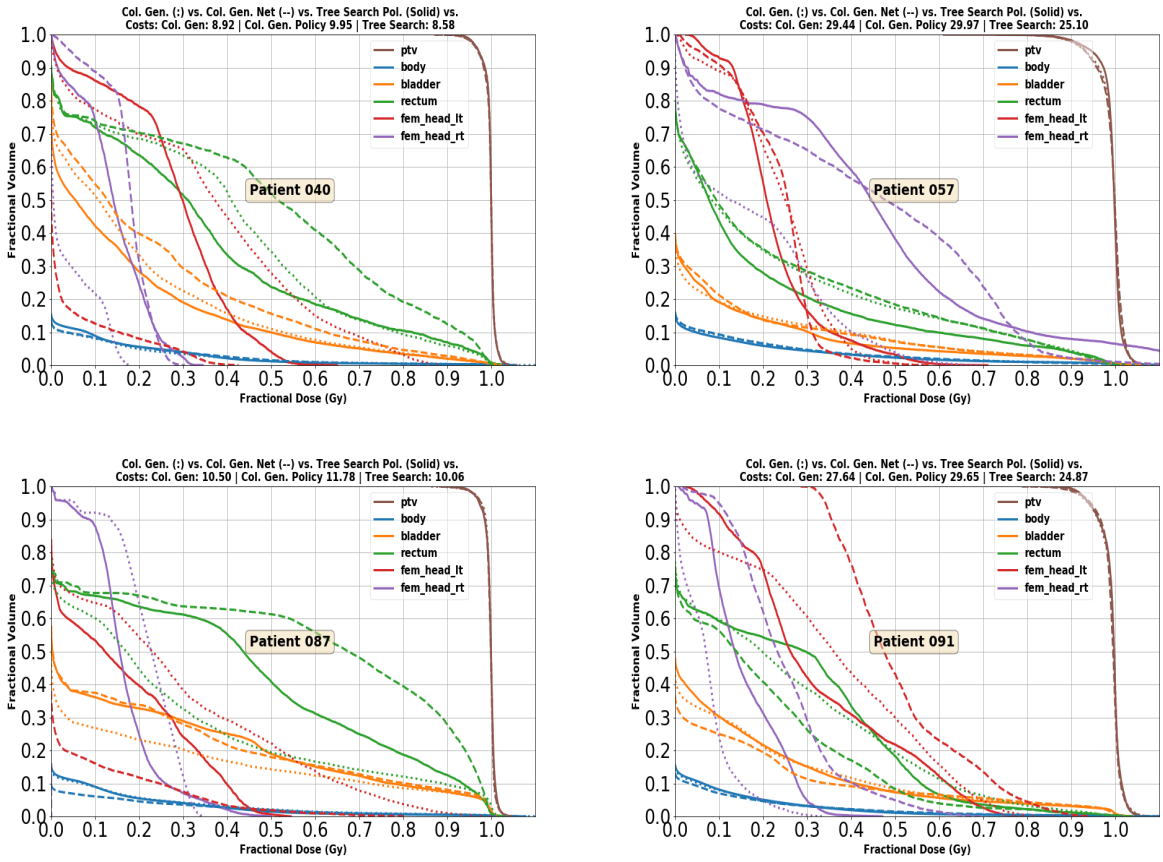


Figure 6.5: DVH curves of our policy versus other schemes.

parameterized CG of (Sadeghnejad Barkousaraie et al., 2019a). On a DVH plot, we expect that the PTV chart would have a dose level of near or equal to $1.0Gy$ while all OARs will ideally have a $0.0Gy$ dose level across all fractional organ volumes. Observing the plots above, we see that the tree search algorithm generally finds beam angles that outperform the base policy network and original column generation linear programming algorithm in terms of lower FMO costs, and better organs' sparing. On average, the time to predict beam angles on an NVIDIA GTX1080 GPU was roughly $48 - 75$ seconds. Similarly, the dose distribution charts of Figure 6.3 present a qualitative evaluation of the impact of the beam angles on the PTV, OARs and healthy tissues. In general, we would like selected beams to damage all PTV cells while ideally sparing all OARs. Given the distribution plots of Figure 6.3, we notice a nice spread of the angles chosen for the prostate test cases. Due to the

symmetrical anatomical nature of the patient's body, we conjecture that the scheme selecting two angles that are geometrically opposite on the patient's body may result in multi-local optima doses.

6.5 Conclusion

In this chapter, we presented a rollout tree search approach for simulating large beam plans for myriad patients with supervised pre-training of a base-policy to guide our tree search. We used a state-action value function that enabled the estimation of a node's value; this relied on a fast Chambolle-Pock-based fluence map optimization of a node's beamlets. With the optimal cost of this fluence cost, we continually backed up the values of a node to its root. To mitigate angle collisions, we re-introduced the pairwise distance scheme of (Ogunmolu et al., 2018). This yielded a balanced spread in angles at each node of the tree such that when the resulting tree policy was parameterized by a neural network policy, it yields a good spread of beam plans. Contrary to a majority of current approaches, our scheme produces beam plans in at most 45 seconds for a majority of the prostate test cases. This is a significant savings in time and effort compared to the manual beam tuning scheme employed in a majority of clinics. Future work will incorporate robustness mechanisms into the approximated mapping from patient geometries to beam angles.

It begs to ask the question, *"why use the neural network base policy to warmstart the search?" "Why not just randomly start from any beam angle?"* Suppose we randomly initiate the beams at a root node, the search may wander off in any direction in the tree. This would elongates the search process. The policy evaluation network earlier proposed (Sadeghnejad Barkousaraie et al., 2019) $i.e.$ $\{\mu_0, \ldots, \mu_{N-1}\}$ can be thought of as a guided policy network that guides the tree search along promising paths of the states exploration space. Another question that comes to mind is *to what extent does the length of the game influence the quality of the tree policy?* General tree search policies rely on the central limit theorem: the mean of N independent realizations of a random variable with mean μ and variance σ^2 would tend towards a normal distribution with mean μ and variance $\frac{\sigma^2}{N}$. Therefore, we have stretched out length of the games in this work to allow as close a

realization as possible to the fact that the expected value of the cost Q would be lower than if the game were abruptly stopped.

Given the increasing popularity of volumetric modulated arc therapy (VMAT) in cancer treatment that arises from its high plan quality and efficiency (Hoffmann et al., 2019; Quan et al., 2012), the clinical application of new methods for standard IMRT treatment planning processes, including BOO, may seem less appealing. However, the recent use of highly non-coplanar plans, such as 4π radiotherapy (Nguyen et al., 2016), and Station Parameter Optimized Radiation Therapy (SPORT) (Dong et al., 2018; Kim et al., 2015), have made finding a fast method for beam orientation selection problem a research problem that is worth revisiting. For a clinical evaluation of the proposed method, we plan to further extend the model for non-coplanar treatment planning features. Given the success of CG in highly non-coplanar problems (OConnor et al., 2018), (Dong et al., 2013), (Dong et al., 2013), (Woods et al., 2016), we expect that this scheme will efficiently approximate the non-coplanar BOO solution.

CHAPTER 7

CONCLUSIONS AND FUTURE WORK

Patient immobilization with frames and masks in radiation therapy achieves the immobilization requirement fairly well. However, these rigid immobilization devices have the disadvantage that they attenuate radio/ionizing radiation, and impose claustrophobia in the patient's head and neck space. Development of frameless and maskless radiation therapy are at the infant stages particularly in real-time feedback motion control at research centers (Belcher, 2017; Haas et al., 2012; Wiersma et al., 2009), (Herrmann et al., 2011), (Liu et al., 2015). We have presented a 1-DOF , a 3-DOF , and a soft robot mechanism capable of 6-DOF head and neck manipulation for frameless and maskless immobilization. In addition to these, we analyzed the deformation, kinematics, and dynamics of spherically symmetric soft robots arranged around the head and neck region of a patient. These model-based approaches were synthesized using finite elastic deformations and the established theory in the kinematics of contact and grasp. In the future, we hope to implement, verify, and validate the accuracy of these proposed mechanisms. In a manner consistent with the full treatment platform such as used in Cyberknife or Novalis ExacTrac treatment facilities, we expect that our design should be compatible to these systems such that it can accommodate a complete positioning correction with radio transparent immobilization devices.

Additionally, we presented an approximate dynamic programming approach for predicting feasible beams in the inverse treatment planning problem at radiation oncology clinics. The chief advantage of our method, compared to current schemes is the time it takes to predict beams on a given patient's CT geometry. In the future, we would like to verify the viability of our method in a clinical setting, and comparing our approach to VMAT predictions.

APPENDIX A

DERIVATION OF IAB-HEAD CONTACT KINEMATICS

A.1 Contact Coordinates and Gaussian Map

Here, we formulate the contact kinematics between an IAB and the head in a fashion similar to the single finger soft contact type postulated in (Murray and Sastry, 1990). We note that an alternative derivation that is more concise can be found in (Montana, 1988). Following equations (3.16a), (3.16b), and 3.17, we write

$$R\, f_1(\alpha_1) + p = f_h(\alpha_h) \tag{A.1a}$$

$$R\, n_1(\alpha_1) = -n_h(\alpha_h) \tag{A.1b}$$

$$R\, \frac{\partial f_1}{\partial \alpha_1}\, M_1^{-1}\, R_\psi = \frac{\partial f_h}{\partial \alpha_h}\, M_h^{-1}. \tag{A.1c}$$

Differentiating (A.1a) and (A.1b), we find that

$$\dot{R}\, f_1(\alpha_1) + R\, \frac{\partial f_1}{\partial \alpha_1}\dot{\alpha_1} + \dot{p} = \frac{\partial f_h}{\partial \alpha_h}\dot{\alpha_h} \tag{A.2}$$

$$\dot{R}\, n_1(\alpha_1) + R\, \frac{\partial n_1}{\partial \alpha_1}\dot{\alpha_1} = -\frac{\partial n_h}{\partial \alpha_h}\dot{\alpha}_h. \tag{A.3}$$

It follows through the multiplication of (A.2) by $\frac{\partial f_h}{\partial \alpha_h}^T$ and putting α_h into (A.3), we have

$$\dot{R}\, n_1(\alpha_1) + R\, \frac{\partial n_1}{\partial \alpha_1}\dot{\alpha_1} = -\frac{\partial n_h}{\partial \alpha_h} M_h^{-2} \frac{\partial f_h}{\partial \alpha_h}^T \left(\dot{R}\, f_1(\alpha_1) + R\, \frac{\partial f_1}{\partial \alpha_1}\dot{\alpha_1} + \dot{p} \right). \tag{A.4}$$

Now, putting (A.1c) into (A.4) and rearranging, we find that

$$\left[R\, \frac{\partial n_1}{\partial \alpha_1} + \frac{\partial n_h}{\partial \alpha_h} M_h^{-2} \left(\frac{\partial f_h^T}{\partial \alpha_h}\, \frac{\partial f_h}{\partial \alpha_h} \right)\, M_h^{-1}\, R_\psi M_1 \right] \dot{\alpha_1} = -\dot{R}\, n_1 - \frac{\partial n_h}{\partial \alpha_h} M_h^{-2} \frac{\partial f_h}{\partial \alpha_h}^T \left(\dot{R}\, f_1(\alpha_1) + \dot{p} \right). \tag{A.5}$$

Multiplying throughout by $M_h^{-T} \frac{\partial f_h}{\partial \alpha_h}^T$, we have on the left hand side of the above as,

$$M_h^{-T} \frac{\partial f_h}{\partial \alpha_h}^T \left(R\, \frac{\partial n_1}{\partial \alpha_1} + \frac{\partial n_h}{\partial \alpha_h}\, M_h^{-1}\, R_\psi M_1 \right) \dot{\alpha}_1. \tag{A.6}$$

Since

$$M_h^{-T}\frac{\partial f_h}{\partial \alpha_h}^T = \frac{\partial f_h}{\partial \alpha_h}M_h^{-1} = \left(R\,\frac{\partial f_1}{\partial \alpha_1}\,M_1^{-1}\,R_\psi\right)^T = R_\psi\,M_1^{-T}\,\frac{\partial f_1}{\partial \alpha_1}^T\,R^T, \tag{A.7}$$

equation (A.6) becomes

$$\begin{aligned}&\left(R_\psi\,M_1^{-T}\,\frac{\partial f_1}{\partial \alpha_1}^T\,\frac{\partial n_1}{\partial \alpha_1} + M_h^{-T}\frac{\partial f_h}{\partial \alpha_h}^T\frac{\partial n_h}{\partial \alpha_h}\,M_h^{-1}\,R_\psi M_1\right)\dot{\alpha}_1\\ &= \left(R_\psi\,M_1^{-T}\,\frac{\partial f_1}{\partial \alpha_1}^T\,\frac{\partial n_1}{\partial \alpha_1}M_1^{-1}R_\psi + M_h^{-T}\frac{\partial f_h}{\partial \alpha_h}^T\frac{\partial n_h}{\partial \alpha_h}\,M_h^{-1}\right)\,R_\psi M_1\dot{\alpha}_1.\end{aligned} \tag{A.8}$$

Setting

$$\tilde{\mathcal{K}}_1 = R_\psi\,M_1^{-T}\,\frac{\partial f_1}{\partial \alpha_1}^T\,\frac{\partial n_1}{\partial \alpha_1}M_1^{-1}R_\psi \quad \text{and } \mathcal{K}_h = M_h^{-T}\frac{\partial f_h}{\partial \alpha_h}^T\frac{\partial n_h}{\partial \alpha_h}\,M_h^{-1},$$

it follows from (A.4) that

$$\begin{aligned}\left(\tilde{\mathcal{K}}_1 + \mathcal{K}_h\right)R_\psi M_1\dot{\alpha}_1 &= M_h^{-T}\frac{\partial f_h}{\partial \alpha_h}^T\left[-\dot{R}\,n_1 - \frac{\partial n_h}{\partial \alpha_h}M_h^{-2}\frac{\partial f_h}{\partial \alpha_h}^T\left(\dot{R}\,f_1 + \dot{p}\right)\right]\\ &= -M_h^{-T}\frac{\partial f_h}{\partial \alpha_h}^T\dot{R}\,n_1 - \mathcal{K}_h\,M_h^{-T}\frac{\partial f_h}{\partial \alpha_h}^T\left(\dot{R}\,f_1 + \dot{p}\right)\end{aligned} \tag{A.9}$$

so that

$$\dot{\alpha}_1 = \left(\tilde{\mathcal{K}}_1 + \mathcal{K}_h\right)^{-1}R_\psi M_1^{-1}\left[\underbrace{-M_h^{-T}\frac{\partial f_h}{\partial \alpha_h}^T\dot{R}\,n_1}_{w_t} - \mathcal{K}_h\,\underbrace{M_h^{-T}\frac{\partial f_h}{\partial \alpha_h}^T\left(\dot{R}\,f_1 + \dot{p}\right)}_{v_t}\right] \tag{A.10}$$

or

$$\dot{\alpha}_1 = \left(\tilde{\mathcal{K}}_1 + \mathcal{K}_h\right)^{-1}R_\psi M_1^{-1}(w_t - \mathcal{K}_h v_t). \tag{A.11}$$

Finding the generalized velocity of the head with respect to a single soft robot's deformation is tantamount to finding $(\hat{w}, v) = \dot{g}\,g^{-1}$. Thus, we find that

$$\omega_t = -M_h^{-T}\frac{\partial f_h}{\partial \alpha_h}^T\left(\omega \times (R\,n_1)\right) = -M_h^{-T}\frac{\partial f_h}{\partial \alpha_h}^T\left(n_h \times \omega\right) \tag{A.12}$$

$$v_t = M_h^{-T}\frac{\partial f_h}{\partial \alpha_h}^T\left(\omega \times (R\,f_1) + \omega \times p + v\right) \tag{A.13}$$

$$= M_h^{-T}\frac{\partial f_h}{\partial \alpha_h}^T\left(-f_h \times \omega + v\right), \tag{A.14}$$

where ω_t is the head's rolling velocity projected onto the contact's tangent plane. The rotation normal to the surface is canceled by the cross product of ω and n_h. In the same vein, v_t is the sliding velocity between the contacts, projected onto the tangent plane. Following the above construction, we find the kinematics of the contact point of the head in local coordinates is

$$\dot{\alpha}_h = M_h^{-1} \left(\tilde{\mathcal{K}}_1 + \mathcal{K}_h \right)^{-1} (\omega_t - \tilde{\mathcal{K}}_1 v_t), \tag{A.15}$$

where $\left(\tilde{\mathcal{K}}_1 + \mathcal{K}_h \right)$ is the *relative curvature* (Montana, 1988). It remains to solve for the relative orientation between the two local coordinates, ψ.

A.2 Relative Contact Orientation and Torsion Metric Tensors

In matrix form, (A.1b) and (A.1c) can be written as,

$$R \begin{bmatrix} \frac{\partial f_1}{\partial \alpha_1} M_1^{-1} & n_1(\alpha_1) \end{bmatrix} \begin{bmatrix} R_\psi & 0 \\ 0 & -1 \end{bmatrix} = \begin{bmatrix} \frac{\partial f_h}{\partial \alpha_h} M_h^{-1} & n_h(\alpha_h) \end{bmatrix}. \tag{A.16}$$

Following the normalized Gaussian frame defined in (3.20), we can rewrite the above equation as

$$R[x_1\, y_1\, z_1]\bar{R}_\psi = [x_h\, y_h\, z_h]. \tag{A.17}$$

The total derivative of (A.17) yield

$$\dot{R} \begin{bmatrix} x_1 & y_1 & z_1 \end{bmatrix} \bar{R}_\psi + R \begin{bmatrix} \dot{x}_1 & \dot{y}_1 & \dot{z}_1 \end{bmatrix} \bar{R}_\psi + R \begin{bmatrix} x_1 & y_1 & z_1 \end{bmatrix} \begin{bmatrix} \dot{R}_\psi & 0 \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \dot{x}_h & \dot{y}_h & \dot{z}_h \end{bmatrix}. \tag{A.18}$$

Premultiplying by $y_1^T R^T$ and then postmultiplying by $\bar{R}_\psi \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}$, with the knowledge that $\bar{R}_\psi \bar{R}_\psi =$ **I**, and the identity $y_1^T y_1 = 1$, we find that

$$y_1^T R^T \dot{R}[x_1 \ y_1 \ z_1]\bar{R}_\psi + y_1^T[\dot{x}_1, \ \dot{y}_1, \ \dot{z}_1]\bar{R}_\psi + (0 \ 1 \ 0) \begin{bmatrix} \dot{R}_\psi & 0 \\ 0 & 0 \end{bmatrix} = y_1^T R^T [\dot{x}_h \ \dot{y}_h \ \dot{z}_h] \tag{A.19}$$

$$y_1^T R^T \dot{R} x_1 + y_1^T \dot{x}_1 + (0 \ 1 \ 0) \begin{bmatrix} \dot{R}_\psi R_\psi & 0 \\ 0 & 0 \end{bmatrix} \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} = y_1^T R^T [\dot{x}_h \ \dot{y}_h \ \dot{z}_h] \bar{R}_\psi \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \tag{A.20}$$

$$y_1^T R^T \dot{R} x_1 + y_1^T \dot{x}_1 + (0 \quad 1) \begin{bmatrix} 0 & \dot{\psi} \\ -\dot{\psi} & 0 \end{bmatrix} \begin{pmatrix} 1 \\ 0 \end{pmatrix} = y_1^T R^T [\dot{x}_h \ \dot{y}_h \ \dot{z}_h] \bar{R}_\psi \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \tag{A.21}$$

$$y_1^T R^T \dot{R} x_1 + y_1^T \dot{x}_1 - \dot{\psi} = y_1^T R^T [\dot{x}_h \ \dot{y}_h \ \dot{z}_h] \bar{R}_\psi \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}. \tag{A.22}$$

From (A.17), we have that

$$\bar{R}_\psi^T \begin{bmatrix} x_1^T & y_1^T & z_1^T \end{bmatrix} R^T = \begin{bmatrix} x_h^T & y_h^T & z_h^T \end{bmatrix} \tag{A.23}$$

so that

$$\begin{bmatrix} x_1^T & y_1^T & z_1^T \end{bmatrix} R^T = \bar{R}_\psi \begin{bmatrix} x_h^T & y_h^T & z_h^T \end{bmatrix} \tag{A.24}$$

or

$$y_1^T R^T = (0 \ 1 \ 0) \, \bar{R}_\psi \begin{bmatrix} x_h^T & y_h^T & z_h^T \end{bmatrix} = (0 \ 1) \, R_\psi \begin{pmatrix} x_h^T \\ y_h^T \end{pmatrix}. \tag{A.25}$$

It follows from (A.22) that

$$\dot{\psi} = y_1^T R^T \dot{R} x_1 + y_1^T \frac{\partial x_1}{\partial \alpha_1} \dot{\alpha}_1 - (0,\, 1)\, R_\psi \begin{bmatrix} x_h^T \dot{x}_h & x_h^T \dot{y}_h \\ y_h^T \dot{x}_h & y_h^T \dot{y}_h \end{bmatrix} R_\psi \begin{pmatrix} 1 \\ 0 \end{pmatrix}. \tag{A.26}$$

Using the identities,

$$x_i^T y_i = 0, \implies \dot{x}_i^T y_i = -x_i^T \dot{y}_i = y_i^T \dot{x}_i \tag{A.27}$$

$$x_i^T x_i = 1, \implies \dot{x}_i^T x_i = 0, \tag{A.28}$$

we can rewrite (A.26) as

$$\begin{aligned} \dot{\psi} &= y_1^T\, R^T \dot{R}\, x_1 + y_h^T \frac{\partial x_h}{\partial \alpha_h}\, \dot{\alpha}_h + y_1^T \frac{\partial x_1}{\partial \alpha_1}\, \dot{\alpha}_1 \\ &= \omega_n + T_h\, M_h\, \dot{\alpha}_h + T_1\, M_1 \dot{\alpha}_1 \end{aligned} \tag{A.29}$$

where

$$\begin{aligned} \omega_n &= y_1^T\, R^T\, \dot{R} x_1 = (R\, y_1)^T\, w \times (R x_1) \\ &= (R\, z_1)^T\, \omega = z_h^T\, \omega \end{aligned} \tag{A.30}$$

$$T_h = y_h \frac{\partial x_h^T}{\partial \alpha_h} M_h^{-T}, \qquad T_1 = y_1 \frac{\partial x_1^T}{\partial \alpha_1} M_1^{-T}. \tag{A.31}$$

It follows that the first, second and third equations of contact are given by (A.11), (A.15), and (A.29) respectively, $i.e.$

Equations of Contact

$$\dot{\alpha}_1 = \left(\tilde{\mathcal{K}}_1 + \mathcal{K}_h\right)^{-1} R_\psi M_1^{-1} (\omega_t - \mathcal{K}_h v_t) \tag{A.32a}$$

$$\dot{\alpha}_h = M_h^{-1} \left(\tilde{\mathcal{K}}_1 + \mathcal{K}_h\right)^{-1} (\omega_t - \tilde{\mathcal{K}}_1 v_t) \tag{A.32b}$$

$$\dot{\psi} = \omega_n + T_h\, M_h\, \dot{\alpha}_h + T_1\, M_1 \dot{\alpha}_1. \tag{A.32c}$$

APPENDIX B

IAB DYNAMICS

We now derive the overall dynamics for the elastic IAB in Eulerian form. Following (2.21), a point on the surface of the IAB has the following Cartesian description

$$\mathbf{r} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} r\cos\theta\,\sin\phi \\ r\sin\theta\,\sin\phi \\ r\cos\phi \end{bmatrix}, \quad \dot{\mathbf{r}} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} \dot{r}\cos\theta\,\sin\phi - r\dot{\theta}\,\sin\theta\,\sin\phi + r\dot{\phi}\,\cos\theta\,\cos\phi \\ \dot{r}\sin\theta\,\sin\phi + r\dot{\theta}\,\cos\theta\,\sin\phi + r\dot{\phi}\,\sin\theta\,\cos\phi \\ \dot{r}\,\cos\phi - r\,\dot{\phi}\sin\phi \end{bmatrix} \tag{B.1}$$

and the components of $\ddot{\mathbf{r}}$ are

$$\begin{aligned}
\ddot{x} &= \cos\theta\left(2\dot{r}\dot{\phi}\cos\theta + r\cos\theta\ddot{\phi} - 2r\dot{\theta}\dot{\phi}\sin\theta\right) - \sin\phi\left(\cos\theta\left(-\ddot{r} + r\dot{\theta}^2 + r\dot{\phi}^2\right) + \sin\theta\left(2\dot{r}\dot{\theta} + r\ddot{\theta}\right)\right) \\
\ddot{y} &= \cos\phi\left(2r\dot{\theta}\dot{\phi}\cos\theta + \left(2\dot{r}\dot{\phi} + r\ddot{\phi}\right)\sin\theta\right) + \sin\phi\left(2\dot{r}\dot{\theta}\cos\theta + r\ddot{\theta}\cos\theta + \sin\theta\left(\ddot{r} - r\dot{\theta}^2 - r\dot{\phi}^2\right)\right) \\
\ddot{z} &= \cos\phi\left(\ddot{r} - r\dot{\phi}^2\right) - \sin\phi\left(2\dot{r}\dot{\phi} + r\ddot{\phi}\right).
\end{aligned} \tag{B.2}$$

Recall the kinetic energy form (Chapter 3) of a continuum body

$$T = \frac{1}{2}\rho\mathbf{v}(\mathbf{r},t)\cdot\mathbf{v}(\mathbf{r},t) = \frac{1}{2}\rho\|\dot{\mathbf{r}}\|^2. \tag{B.3}$$

The constitutive equation that governs the Cauchy stress tensor, $\boldsymbol{\sigma}$ (2.32), is independent of the path of the deformation from the reference configuration and it is solely a function of the state of deformation; we therefore conclude that the IAB material is Cauchy elastic (Ogden, 1997, §4.2). We therefore choose $V = 0$ following (Ogden, 1997)'s recommendation. Note that since we are treating an incompressible material, the material mass density is uniform throughout the body in its configuration. Thus the rate of change of ρ vanishes.

$$T = \frac{1}{2}\rho\|\dot{\mathbf{r}}\|^2 = \frac{1}{2}\rho\left(\dot{r}^2 + r^2\dot{\phi}^2 + r^2\dot{\theta}^2\sin^2\phi\right), \quad V = 0. \tag{B.4}$$

It follows that the Lagrangian is

$$L(\mathbf{r},\dot{\mathbf{r}}) = \frac{1}{2}\rho\left(\dot{r}^2 + r^2\dot{\phi}^2 + r^2\dot{\theta}^2\sin^2\phi\right). \tag{B.5}$$

The derivatives of the canonical momenta are

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{r}} = \frac{d}{dt}(\rho\,\dot{r}) = \rho\,\ddot{r} \tag{B.6a}$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}} = \frac{d}{dt}\left(r^2\,\rho\,\dot{\theta}\,\sin^2\phi\right) = 2\,\rho\,r\,\dot{r}\,\dot{\theta}\,\sin^2\phi + 2\,\rho\,r^2\,\dot{\phi}\,\dot{\theta}\,\sin\phi\cos\phi + \rho\,r^2\,\ddot{\theta}\sin^2\phi \tag{B.6b}$$

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\phi}} = \frac{d}{dt}\left(r^2\,\rho\,\dot{\phi}\right) = 2\,\rho\,\dot{r}\,r\,\dot{\phi} + \rho\,r^2\,\ddot{\phi} \tag{B.6c}$$

with associated generalized forces

$$\frac{\partial L}{\partial r} = \rho\,r\,\dot{\phi}^2 + \rho\,r\,\dot{\theta}^2\,\sin^2\phi, \quad \frac{\partial L}{\partial \theta} = 0, \quad \frac{\partial L}{\partial \phi} = \rho\,r^2\,\dot{\theta}^2\,\cos\phi\,\sin\phi \tag{B.7a}$$

so that we may write the general system dynamics as

$$\boldsymbol{\tau} = \rho\left\{\ddot{r} + r\left[r\ddot{\phi} + 2\dot{r}\left(\dot{\phi} + \dot{\theta}\sin^2\phi\right) + \sin\phi\left(r\ddot{\theta}\sin\phi - \dot{\theta}^2(r\cos\phi + \sin\phi\right) + \dot{\phi}\left(-\dot{\phi} + r\dot{\theta}\sin 2\phi\right)\right]\right\} \tag{B.8}$$

or in matrix form

$$\boldsymbol{\tau} = \begin{bmatrix} \rho & 0 & 0 \\ 0 & \rho\,r^2 & 0 \\ 0 & 0 & \rho\,r^2\sin^2\phi \end{bmatrix}\begin{bmatrix} \ddot{r} \\ \ddot{\phi} \\ \ddot{\theta} \end{bmatrix} + \mathbf{diag}\begin{bmatrix} 2\,\rho\,r\,\left(\dot{\theta}\,\sin^2\phi + \dot{\phi}\right) \\ \rho r\left(r\dot{\theta}\sin 2\phi - \dot{\phi}\right) \\ -\rho r\dot{\theta}\sin\phi\,(r\cos\phi + \sin\phi) \end{bmatrix}\begin{bmatrix} \dot{r} \\ \dot{\phi} \\ \dot{\theta} \end{bmatrix}. \tag{B.9}$$

APPENDIX C

CONTACT-BASED BOUNDARY VALUE PROBLEM

We make the simplifying assumption that the deformation is spherically symmetric, and we expect that the shear stress contributions $\sigma_{r\phi}, \sigma_{r\theta}, \sigma_{\phi\theta}$ vanish in (3.6) so that we have

$$\frac{1}{r^2}\frac{\partial}{\partial r}(r^2\sigma_{rr}) - \frac{1}{r}(\sigma_{\theta\theta} + \sigma_{\phi\phi}) + \rho b_r = \rho\ddot{\mathbf{r}}_x \tag{C.1a}$$

$$\frac{1}{r\sin\phi}\frac{\partial}{\partial\phi}(\sin\phi\sigma_{\phi\phi}) - \frac{\cot\phi}{r}(\sigma_{\theta\theta}) + \rho b_\phi = \rho\ddot{\mathbf{r}}_y \tag{C.1b}$$

$$\frac{1}{r\sin\phi}\frac{\partial}{\partial\theta}(\sigma_{\theta\theta}) + \rho b_\theta = \rho\ddot{\mathbf{r}}_z \tag{C.1c}$$

where the mass density ρ is uniform throughout the body and the components $\ddot{\mathbf{r}}_x$,$\ddot{\mathbf{r}}_y$, and $\ddot{\mathbf{r}}_z$ are as given in (B.6). Solving the equations in (C.1), we have from (C.1a),

$$\begin{aligned}\frac{\partial\sigma_{rr}}{\partial r} &= \rho\ddot{\mathbf{r}}_x + \frac{1}{r}(\sigma_{\theta\theta} + \sigma_{\phi\phi}) - \rho b_r \\ &= \frac{1}{r}(\sigma_{\theta\theta} + \sigma_{\phi\phi}) - \rho b_r + \rho\cos\theta\left(2\dot{r}\dot{\phi}\cos\theta + r\cos\theta\ddot{\phi} - 2r\dot{\theta}\dot{\phi}\sin\theta\right) - \\ &\qquad \rho\sin\phi\left(\cos\theta(-\ddot{r} + r\dot{\theta}^2 + r\dot{\phi}^2) + \sin\theta(2\dot{r}\dot{\theta} + r\ddot{\theta})\right)\end{aligned} \tag{C.2}$$

and from (C.1b), we have

$$\rho\ddot{\mathbf{r}}_y = \cancelto{0}{\frac{\cot\phi}{r}(\sigma_{\phi\phi} - \sigma_{\theta\theta})} + \rho b_\phi + \frac{1}{r}\frac{\partial\sigma_{\phi\phi}}{\partial\phi} \implies \frac{\partial\sigma_{\phi\phi}}{\partial\phi} = r\rho\ddot{\mathbf{r}}_y - r\rho b_\phi.$$

Thus,

$$\begin{aligned}\frac{\partial\sigma_{\phi\phi}}{\partial\phi} = -\rho r b_\theta + r\rho\Big[&\cos\phi\left(2r\dot{\theta}\dot{\phi}\cos\theta + (2\dot{r}\dot{\phi} + r\ddot{\phi})\sin\theta\right) + \\ &\sin\theta\left(2\dot{r}\dot{\theta}\cos\theta + r\ddot{\theta}\cos\theta + (\ddot{r} - r\dot{\theta}^2 - r\dot{\phi}^2)\right)\sin\phi\Big]\end{aligned} \tag{C.3}$$

and lastly, we have from (C.1c)

$$\frac{\partial\sigma_{\theta\theta}}{\partial\theta} = (\rho\ddot{\mathbf{r}}_z - \rho b_\theta)r\sin\phi = -r\rho b_\theta\sin\phi + r\rho\sin\phi\cos\phi\left(\ddot{r} - r\dot{\phi}^2\right) - r\rho\sin^2\phi\left(2\dot{r}\dot{\phi} + r\ddot{\phi}\right). \tag{C.4}$$

Collecting $\frac{\partial \sigma_{rr}}{\partial r}$, $\frac{\partial \sigma_{\theta\theta}}{\partial \theta}$, and $\frac{\partial \sigma_{\phi\phi}}{\partial \phi}$ above, taking $\sigma_{rr}(r_\circ) = \sigma_{\theta\theta}(2\pi) = \sigma_{\phi\phi}(\pi) = 0$ and integrating from the internal to outer boundary conditions of (2.19), we have the full forms of the normal stress components as

$$\sigma_{rr}(\delta) = -\int_{\delta}^{r_\circ} \left[\frac{1}{r}\left(-2p + 2C_1\frac{r^2}{R^2} - 2C_2\frac{R^8}{r^8}\right) - \rho b_r + \rho\cos\theta\left(2\dot{r}\dot{\phi}\cos\theta + r\cos\theta\ddot{\phi} - 2r\dot{\theta}\dot{\phi}\sin\theta\right)\right.$$
$$\left. -\rho\sin\phi\left(\cos\theta(-\ddot{r} + r\dot{\theta}^2 + r\dot{\phi}^2) + \sin\theta(2\dot{r}\dot{\theta} + r\ddot{\theta})\right)\right] dr, \text{ where } r_i \leq \delta \leq r_\circ \quad \text{(C.5a)}$$

$$\sigma_{\phi\phi}(\epsilon) = -\int_{\epsilon}^{\pi} \left[r\rho\left[\cos\phi\left(2r\dot{\theta}\dot{\phi}\cos\theta + (2\dot{r}\dot{\phi} + r\ddot{\phi})\sin\theta\right) + \right.\right.$$
$$\left.\left. \sin\theta\left(2\dot{r}\dot{\theta}\cos\theta + r\ddot{\theta}\cos\theta + (\ddot{r} - r\dot{\theta}^2 - r\dot{\phi}^2)\right)\sin\phi\right] - \rho r b_\theta\right] d\phi,$$
$$\text{where } 0 \leq \epsilon \leq \pi \quad \text{(C.5b)}$$

$$\sigma_{\theta\theta}(\zeta) = -\int_{\zeta}^{2\pi} \left[-r\rho b_\theta\sin\phi + r\rho\sin\phi\cos\phi\left(\ddot{r} - r\dot{\phi}^2\right) - r\rho\sin^2\phi\left(2\dot{r}\dot{\phi} + r\ddot{\phi}\right)\right] d\theta,$$
$$\text{where } 0 \leq \zeta \leq 2\pi. \quad \text{(C.5c)}$$

Now, using the boundary condition, $P + P_{\text{atm}} = -\sigma_{rr}|_{r=r_i}$, where P_{atm} is the atmospheric pressure, (here, taken as 0), we have the internal pressure in the IAB cavity as a function of the radius of deformation in the current configuration as

$$P = \int_{r_i}^{r_\circ} \left[\frac{1}{r}\left(-2p + 2C_1\frac{r^2}{R^2} - 2C_2\frac{R^8}{r^8}\right) - \rho b_r + \rho\cos\theta\left(2\dot{r}\dot{\phi}\cos\theta + r\cos\theta\ddot{\phi} - 2r\dot{\theta}\dot{\phi}\sin\theta\right)\right.$$
$$\left. -\rho\sin\phi\left(\cos\theta(-\ddot{r} + r\dot{\theta}^2 + r\dot{\phi}^2) + \sin\theta(2\dot{r}\dot{\theta} + r\ddot{\theta})\right)\right] dr. \quad \text{(C.6)}$$

APPENDIX D

ADMM UPDATE EQUATIONS

We can rewrite the above objective of (5.1), subject to nonnegative bixel intensity constraints, as the minimization problem

$$\min \frac{1}{2}\|A\mathbf{x} - \mathbf{b}\|_2^2 \quad \text{subject to } x \geq 0.$$

The Lagrangian thus becomes

$$L(\boldsymbol{x}, \boldsymbol{\lambda}) = \frac{1}{2}\|A\mathbf{x} - \mathbf{b}\|_2^2 - \boldsymbol{\lambda}^T\boldsymbol{x},$$

where $\boldsymbol{\lambda} \in \mathbb{R}^n$ is a multiplier. This problem can be solved with dual gradient descent (DGD), but DGD has the drawback that the primal and dual updates are not robust to objective's constraints (Boyd et al., 2011). The alternating direction method of multipliers (ADMM) (Boyd et al., 2011) tackles the robustness problem by adding a quadratic penalty term to the Lagrangian and alternatingly updating the $\boldsymbol{x}$ and $\boldsymbol{\lambda}$ variables in a "broadcast and gather" process. This turns out to be attractive since we will be solving a large-scale learning problem for the optimal beam angle set combination. Introducing an auxiliary variable $\mathbf{z}$, we have

$$\min_{\boldsymbol{x}} \frac{1}{2}\|\boldsymbol{A}\boldsymbol{x} - \mathbf{b}\|_2^2, \quad \text{subject to } \boldsymbol{z} = \boldsymbol{x},\ \boldsymbol{z} \geq 0,$$

so that the Lagrangian can be written as,

$$\min_{\boldsymbol{x},\boldsymbol{z}} \frac{1}{2}\|\boldsymbol{A}\boldsymbol{x} - \mathbf{b}\|_2^2 - \boldsymbol{\lambda}^T(\boldsymbol{z} - \boldsymbol{x}) + \frac{\rho}{2}\|\boldsymbol{z} - \boldsymbol{x}\|_2^2, \tag{D.1}$$

where $\rho > 0$ is an ADMM penalty parameter. Minimizing (D.1) w.r.t $\boldsymbol{x}$, the $\boldsymbol{x}$ subproblem of (D.1) yields

$$\min_{\boldsymbol{x}} \quad \frac{1}{2}\boldsymbol{x}^T(\boldsymbol{A}^T\boldsymbol{A} + \rho\boldsymbol{I})\boldsymbol{x} + (\boldsymbol{\lambda}^T - \boldsymbol{A}^T\boldsymbol{b} - \rho\boldsymbol{z}^T)\boldsymbol{x},$$

so that the $\boldsymbol{x}$-update (due to the convex quadratic nature of the problem) becomes,

$$\boldsymbol{x}^{k+1} = \left(\boldsymbol{A}^T\boldsymbol{A} + \rho\boldsymbol{I}\right)^{-1}\left(\boldsymbol{A}^T\boldsymbol{b} + \rho\boldsymbol{z}^k - \boldsymbol{\lambda}^k\right). \tag{D.2}$$

Similarly, the $\boldsymbol{z}$-update for (D.1) can be found by the z-minimization subproblem

$$\min_{z} \; -\boldsymbol{\lambda}^T \boldsymbol{z} + \frac{\rho}{2}\|\boldsymbol{z} - \boldsymbol{x}\|_2^2 := \min_{z} \; \frac{\rho}{2}\|\boldsymbol{z} - \boldsymbol{x} - \frac{1}{\rho}(\boldsymbol{\lambda})\|_2^2.$$

Using the soft-thresholding operator, $S_{\boldsymbol{\lambda}/\rho}$, we find that

$$\boldsymbol{z}^{k+1} = S_{\boldsymbol{\lambda}/\rho}\left(\boldsymbol{x}^{k+1} + \boldsymbol{\lambda}^k\right), \tag{D.3}$$

where $S_{\boldsymbol{\lambda}/\rho}(\tau) = (x - \boldsymbol{\lambda}/\rho)_+ - (-\tau - \boldsymbol{\lambda}/\rho)_+$. $\boldsymbol{\lambda}$ is updated as

$$\boldsymbol{\lambda}^{k+1} = \boldsymbol{\lambda}^k - \gamma(\boldsymbol{z}^{k+1} - \boldsymbol{x}^{k+1}), \tag{D.4}$$

and γ controls the step length. The inverse operation in (D.2) can be carried out with any iterative solver, e.g. conjugate gradient. We use an over-relaxation parameter, $\alpha^k = 1.5$, and set the quadratic penalty to $\rho = 1.5$, in the $\boldsymbol{z}$ and $\boldsymbol{\lambda}$ updates: $\alpha^k \boldsymbol{A}\,\boldsymbol{x}^{k+1} - (1 - \alpha^k)\boldsymbol{z}^k$. The stopping criterion is met when the primal and dual residuals are sufficiently small, $i.e.$

$$r^k = \|\boldsymbol{x}^k - \boldsymbol{z}^k\|_2 \leq \epsilon^{\text{pri}} \;\text{ and }\; s^k = \| - \rho\,(\boldsymbol{z}^{k+1} - \boldsymbol{z}^k)\|_2 \leq \epsilon^{\text{dual}},$$

with,

$$\epsilon^{\text{pri}} = \sqrt{\rho}\epsilon^{\text{abs}} + \epsilon^{\text{rel}} \max\{\|\boldsymbol{x}^k\|_2, \| - \boldsymbol{z}\|_2\}, \text{ and}$$

$$\epsilon^{\text{dual}} = \sqrt{n}\epsilon^{\text{abs}} + \epsilon^{\text{rel}}(\rho\boldsymbol{\lambda}^{\boldsymbol{k}}), \tag{D.5}$$

where $\epsilon^{\text{pri}} > 0$, $\epsilon^{\text{dual}} > 0$ are the primal and dual feasibility tolerances for the primal and dual feasibility conditions (see (Boyd et al., 2011, §3.3)). In this work, we set $\epsilon^{\text{abs}} = 10^{-4}$ and $\epsilon^{\text{rel}} = 10^{-2}$.

REFERENCES

Agrawal, R. (1995). Sample mean based index policies by o (log n) regret for the multi-armed bandit problem. *Advances in Applied Probability 27*(4), 1054–1078. 110

Ahn, P. H. et al. (2009). Random positional variation among the skull, mandible, and cervical spine with treatment progression during head-and-neck radiotherapy. *Int J Radiat Oncol Biol Phys 73*(2), 626–33. 7

Aleman, D. M., A. Kumar, R. K. Ahuja, H. E. Romeijn, and J. F. Dempsey (2008a). Neighborhood Search Approaches to Beam Orientation Optimization i Intensity Modulated Radiation Therapy Treatment Planning. *Journal of Global Optimization 42*(4), 587–607. 21

Aleman, D. M., A. Kumar, R. K. Ahuja, H. E. Romeijn, and J. F. Dempsey (2008b). Neighborhood Search Approaches to Beam Orientation Optimization i Intensity Modulated Radiation Therapy Treatment Planning. *Journal of Global Optimization 42*(4), 587–607. 97, 111

Almubarak, Y., J. Aniket, O. Ogunmolu, X. Gu, S. Jiang, N. Gans, and Y. Tadesse (2018). Design and development of soft robots for head and neck cancer radiotherapy. In *SPIE: Smart Structures + Nondestructive Evaluation*. SPIE. 23

Andersen et al. (2012). Kinect Depth Sensor Evaluation for Computer Vision Applications. Accessed on Feb 23, 2016. 78

Anderson, B. and J. Moore (1990). *Optimal Control: Linear Quadratic Methods*. Prentice Hall, Englewood Cliffs, New Jersey 07632. 79

Astrom, K. (1965). Optimal Control of Markov Processes with Incomplete State Information. *10*, 174–205. 112

Barkousaraie, A. S., O. Ogunmolu, S. Jiang, and D. Nguyen (2019). Using supervised learning and guided monte carlo tree search for beam orientation optimization in radiation therapy. In *Artificial Intelligence in Radiation Therapy. Lecture Notes in Computer Science.*, pp. 1–9. Springer, Cham. 24, 119, 130

Basar, T. and G. J. Olsder (1999). *Dynamic noncooperative game theory*, Volume 23. Siam. 106, 107, 113

Belcher, A. (2017). *Patient Motion Management with 6-DOF Robotics for Frameless and Maskless Stereotactic Radiosurgery*. Ph. D. thesis, The University of Chicago. 13, 135

Bellman, R. (1957). *Dynamic programming*. Princeton University Press. 98

Bengio, Y. et al. (1994). Learning Long-term Dependencies with gradient Descent is Difficult. *IEEE Transactions on Neural Networks*. doi: 10.1109/72.279181. 92

Bern, J. M., G. Kumagai, and S. Coros (2017). Fabrication, modeling, and control of plush robots. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, pp. 3739–3746. IEEE. 18

Bertsekas, D. P. (2013). *Approximate Policy Iteration : A Survey and Some New Methods.* 23, 24, 130

Bertsekas, D. P., D. P. Bertsekas, D. P. Bertsekas, and D. P. Bertsekas (2017a). *Dynamic Programming and Optimal Control*, Volume 1. Athena Scientific Belmont, MA. 23, 97, 121

Bertsekas, D. P., D. P. Bertsekas, D. P. Bertsekas, and D. P. Bertsekas (2017b). *Dynamic Programming and Optimal Control*, Volume 2. Athena Scientific Belmont, MA. 97

Bertsimas, D., V. Cacchiani, D. Craft, and O. Nohadani (2013). A Hybrid Approach To Beam Angle Optimization In Intensity-modulated Radiation Therapy. *Computers & Operations Research 40*(9), 2187–2197. 20

Besl, Paul J.; McKay, N. D. (1992). A Method for Registration of 3D Shapes. 87

Bishop-Moser, J., G. Krishnan, C. Kim, and S. Kota (2012). Design of soft robotic actuators using fluid-filled fiber-reinforced elastomeric enclosures in parallel combinations. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4264–4269. IEEE. 18

Bortfeld, T. and W. Schlegel (1993). Optimization of beam orientations in radiation therapy: Some theoretical considerations. *Physics in Medicine & Biology 38*(2), 291. 21

Boyd, S., N. Parikh, E. Chu, B. Peleato, and J. Eckstein (2011). Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers. *Foundations and Trends in Machine learning 3*(1), 1–122. 105, 145, 146

Boyer, A. L., E. B. Butler, T. A. DiPetrillo, M. J. Engler, B. Fraass, W. Grant, C. C. Ling, D. A. Low, T. R. Mackie, R. Mohan, J. A. Purdy, M. Roach, J. G. Rosenman, L. J. Verhey, R. L. Wong, John W. Cumberlin, H. Stone, and J. R. Palta (2001). Intensity-modulated radiotherapy: current status and issues of interest. *International Journal of Radiation Oncology* Biology* Physics 51*(4), 880–914. 22

Boyer, A. L., T. G. Ochran, C. E. Nyerick, T. J. Waldron, and C. J. Huntzinger (1992). Clinical dosimetry for implementation of a multileaf collimator. *Medical physics 19*(5), 1255–1261. 6

Boyer, F., M. Porez, and W. Khalil (2006). Macro-continuous computed torque algorithm for a three-dimensional eel-like robot. *IEEE Transactions on Robotics 22*(4), 763–775. 18

Brockett, R. W. (1990). Some Mathematical Aspects of Robotics. In *Proceedings of Symposia in Applied Mathematics*, Volume 41, pp. 1–19. 28

Browne, C., E. Powley, D. Whitehouse, S. Lucas, P. I. Cowling, P. Rohlfshagen, S. Tavener, D. Perez, S. Samothrakis, and S. Colton (2012). A Survey of Monte Carlo Tree Search Methods. *IEEE Transactions on Computational Intelligence and AI in games 4*(1). 121

Cacucciolo, V., Y. Ansari, A. L. Shoushtari, M. Cianchetti, and C. Laschi (2015). Adaptive locomotion on uneven terrains by means of a functional separation of time scales in the design and control of robots. *Proceedings of AMAM-Adaptive Motion in Animals and Machines*. 17

Cerviño, L. I., N. Detorie, M. Taylor, J. D. Lawson, T. Harry, K. T. Murphy, A. J. Mundt, S. B. Jiang, and T. A. Pawlicki (2012). Initial Clinical Experience with a Frameless and Maskless Stereotactic Radiosurgery Treatment. *Practical Radiation Oncology 2*(1), 54–62. 12

Cerviño, L. I., T. Pawlicki, J. D. Lawson, and S. B. Jiang (2010). Frame-less and mask-less cranial stereotactic radiosurgery: a feasibility study. *Physics in medicine and biology 55*(7), 1863. 12

Chambolle, A. and T. Pock (2011). A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of mathematical imaging and vision 40*(1), 120–145. 129

Chirikjian, G. S. (1994). Hyper-redundant manipulator dynamics: A continuum approximation. *Advanced Robotics 9*(3), 217–243. 18

Chirikjian, G. S. and J. W. Burdick (1995). The kinematics of hyper-redundant robot locomotion. *IEEE transactions on robotics and automation 11*(6), 781–793. 18

Chung, M., M. Buro, and J. Schaeffer (2005). Monte carlo planning in rts games. In *CIG*. Citeseer. 98

Coulom, R. (2006). Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search. *International Conference on Computers and Games*. 97, 121

Craft, D. (2007). Local beam angle optimization with linear programming and gradient search. *Physics in Medicine & Biology 52*(7), N127. 20

D D'Souza, W., R. R. Meyer, and L. Shi (2004). Selection Of Beam Orientations In Intensity-modulated Radiation Therapy Using Single-beam Indices And Integer Programming. *Physics in Medicine & Biology 49*(15), 3465. 22

Demirkoparan, H. and T. J. Pence (2007). Swelling of an Internally Pressurized Nonlinearly Elastic Tube with Fiber Reinforcing. *International journal of solids and structures 44*(11-12), 4009–4029. 18, 26

Dinh, H., S. Bhasin, R. Kamalapurkar, and W. E. Dixon (2017). Dynamic Neural Network-based Output Feedback Tracking Control for Uncertain Nonlinear Systems. *Journal of Dynamic Systems, Measurement, and Control*. 93

Djajaputra, D., Q. Wu, Y. Wu, and R. Mohan (2003). Algorithm and Performance Of A Clinical Imrt Beam-angle Optimization System. *Physics in Medicine & Biology 48*(19), 3191. 21

Dong, P., P. Lee, D. Ruan, T. Long, E. Romeijn, D. A. Low, P. Kupelian, J. Abraham, Y. Yang, and K. Sheng (2013). 4π noncoplanar stereotactic body radiation therapy for centrally located or larger lung tumors. *International Journal of Radiation Oncology* Biology* Physics 86*(3), 407–413. 129, 134

Dong, P., P. Lee, D. Ruan, T. Long, E. Romeijn, Y. Yang, D. Low, P. Kupelian, and K. Sheng (2013). 4π non-coplanar liver sbrt: a novel delivery technique. *International Journal of Radiation Oncology* Biology* Physics 85*(5), 1360–1366. 129, 134

Dong, P., H. Liu, and L. Xing (2018). Monte Carlo tree search-based non-coplanar trajectory design for station parameter optimized radiation therapy (SPORT). *Physics in Medicine and Biology 63*(13). 134

Durrant-Whyte, H. (2001, Jan). Introduction to Estimation and the Kalman Filter. Accessed Aug 05, 2015. 77

Ensenso. Flexview. `http://www.ensenso.com/products/flexview/`. Accessed on January 21, 2016. 83

Fung, Y.-c., P. Tong, and X. Chen (2001). *Classical and computational solid mechanics*, Volume 2. World Scientific Publishing Company. 38

Gelly, S. and D. Silver (2011). Monte-Carlo tree search and rapid action value estimation in computer Go. *Artificial Intelligence 175*, 1856–1875. 97, 121, 123

Gent, A. (2012). *Engineering with Rubber. How to Design Rubber Components*. Munich: Carl Hanser Verlag Publicationbs. 18, 26, 31, 48

George Thuruthel, T., Y. Ansari, E. Falotico, and C. Laschi (2018). Control Strategies for Soft Robotic Manipulators: A Survey. *Soft Robotics 5*(2), 149–163. 18, 25

Gevaert, T., D. Verellen, K. Tournel, N. Linthout, S. Bral, B. Engels, C. Collen, T. Depuydt, M. Duchateau, T. Reynders, et al. (2012). Setup Accuracy Of The Novalis Exactrac 6dof System For Frameless Radiosurgery. *International Journal of Radiation Oncology* Biology* Physics 82*(5), 1627–1635. 7

Giorelli, M., F. Renda, M. Calisti, A. Arienti, G. Ferri, and C. Laschi (2015). Neural Network And Jacobian Method For Solving The Inverse Statics Of A Cable-driven Soft Arm With Nonconstant Curvature. *IEEE Transactions on Robotics 31*(4), 823–834. 18

Haas, O., G. Bueno, R. Spriestersbach, H. Himmler, K. Burnham, and J. Mills (2005). Imaging And Control For Adaptive Radiotherapy. *IFAC Proceedings Volumes 38*(1), 25–30. 14

Haas, O. C., P. Skworcow, D. Paluszczyszyn, A. Sahih, M. Ruta, and J. A. Mills (2012). Couch-Based Motion Compensation: Modelling, Simulation and Real-time Experiments. *Physics in Medicine & Biology 57*(18), 5787. 14, 135

Hahnloser, R. H., R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, and H. S. Seung (2000). Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. *Nature 405*(6789), 947. 102

Hannan, M. W. and I. D. Walker (2000). Novel Kinematics for Continuum Robots. pp. 227–238. Advances in Robot Kinematics. 25

Hannan, M. W. and I. D. Walker (2003). Kinematics and the Implementation of an Elephant's Trunk Manipulator and Other Continuum Style Robots. *Journal of Robotic Systems 20*(2), 45–63. 25

He, K., X. Zhang, S. Ren, and J. Sun (2015). Delving Deep into Rectifiers: Surpassing Human-level Performance on Imagenet Classification. In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034. 103

He, K., X. Zhang, S. Ren, and J. Sun (2016). Deep Residual Learning For Image Recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778. 99, 102

Heinrich, J., M. Lanctot, and D. Silver (2015). Fictitious self-play in extensive-form games. In *International Conference on Machine Learning*, pp. 805–813. 106, 112

Herrmann, C., L. Ma, and K. Schilling (2011). Model Predictive Control For Tumor Motion Compensation In Robot Assisted Radiotherapy. *IFAC Proceedings Volumes 44*(1), 5968–5973. 13, 135

Hochreiter, S. and J. Schmidhuber (1997). Long Short-Term Memory. *Neural computation 9*(8), 1735–80. 87, 92

Hoffmann, M., J. Pacey, J. Goodworth, A. Laszcyzk, R. Ford, B. Chick, S. Greenham, and J. Westhuyzen (2019). Analysis of a volumetric-modulated arc therapy (vmat) single phase prostate template as a class solution. *Reports of Practical Oncology and Radiotherapy 24*(1), 92–96. 134

Holzapfel, G. A., T. C. Gasser, and R. W. Ogden (2000). A new constitutive framework for arterial wall mechanics and a comparative study of material models. *Journal of elasticity and the physical science of solids 61*(1-3), 1–48. 18, 36

Ilievski, F., A. D. Mazzeo, R. F. Shepherd, X. Chen, and G. M. Whitesides (2011). Titelbild: Soft robotics for chemists (angew. chem. 8/2011). *Angewandte Chemie 123*(8), 1765–1765. 17

Ioffe, S. and C. Szegedy (2015). Batch Normalization: Accelerating Deep Network Training By Reducing Internal Covariate Shift. *arXiv preprint arXiv:1502.03167*. 102

Jaffray, D. (2012). Image-guided radiotherapy: from current concept to future perspectives. *Nat Rev Clin Oncol 9*(12), 688–699. 8

Jia, X., C. Men, Y. Lou, and S. B. Jiang (2011). Beam Orientation Optimization For Intensity Modulated Radiation Therapy Using Adaptive L2,1-minimization. *Physics in Medicine and Biology 56*(19), 6205–6222. 22

Jones, B. A. and I. D. Walker (2006). Kinematics for Multisection Continuum Robots.pdf. *22*(1), 43–55. 25

Kang, H., D. M. Lovelock, E. D. Yorke, S. Kriminiski, N. Lee, and H. I. Amols (2011). Accurate positioning for head and neck cancer patients using 2d and 3d image guidance. *J Appl Clin Med Phys 12*(1), 3270. 8

Kapadia, A. D., K. E. Fry, and I. D. Walker (2014). Empirical investigation of closed-loop control of extensible continuum manipulators. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 329–335. IEEE. 26

Karush, W. (1939). Minima of functions of several variables with inequalities as side constraints. *M. Sc. Dissertation. Dept. of Mathematics, Univ. of Chicago*. 119

Katzschmann, R. K., A. D. Marchese, and D. Rus. Autonomous Object Manipulation Using a Soft Planar Grasping Manipulator. 17

Kaye, A., R. Stepto, W. Work, J. Aleman, and A. Y. Malkin (1998). Definition of terms relating to the non-ultimate mechanical properties of polymers (recommendations 1998). *Pure and applied chemistry 70*(3), 701–754. 34

Khan, F. M., J. P. Gibbons, and P. W. Sperduto (2016). *Khan's Treatment Planning in Radiation Oncology*. Lippincott Williams & Wilkins. 3, 4

Kim, H., R. Li, R. Lee, and L. Xing (2015). Beams-eye-view dosimetrics (BEVD) guided rotational station parameter optimized radiation therapy (SPORT) planning based on reweighted total-variation minimization. *Physics in Medicine and Biology 60*(5). 134

Kocsis, L. and C. Szepesvári (2006). Bandit based Monte-Carlo Planning. *European Conference on Machine Learning*. 97, 121, 122, 123

Krauss, A., S. Nill, M. Tacke, and U. Oelfke (2011). Electromagnetic real-time tumor position monitoring and dynamic multileaf collimator tracking using a siemens 160 mlc: Geometric and dosimetric accuracy of an integrated system. *International Journal of Radiation Oncology* Biology* Physics 79*(2), 579–587. 12

Kuhn, H. and A. Tucker (1951). Nonlinear Programming. *Proceedings of 2nd Berkeley Symposium. Berkeley*. 119

Landau, Y. D. (1979). *Adaptive Control: The Model Reference Approach*. Marcel Dekker, Inc. 91

Laschi, C., M. Cianchetti, B. Mazzolai, L. Margheri, M. Follador, and P. Dario (2012). Soft robot arm inspired by the octopus. *Advanced Robotics 26*(7), 709–727. 17

Lavretsky, E. and K. Wise (2005). *Robust Adaptive Control with Aerospace Applications*. Springer. 91, 93

LeCun, Y., Y. Bengio, and G. Hinton (2015). Deep Learning. *Nature 521*(7553), 436–444. 97

Lee, S., K.-H. Chang, J. B. Shim, Y. Cao, C. K. Lee, S. J. Cho, D. S. Yang, Y. J. Park, W. S. Yoon, and C. Y. Kim (2012). Evaluation Of Mechanical Accuracy For Couch-based Tracking System (CBTS). *Journal of applied clinical medical physics 13*(6), 157–169. 7

Levine, S., C. Finn, T. Darrell, and P. Abbeel (2016). End-to-End Training of Deep Visuomotor Policies. *Journal of Machine Learning Research 17*, 1–40. 23

Li, G., A. Ballangrud, M. Chan, R. Ma, K. Beal, Y. Yamada, T. Chan, J. Lee, P. Parhar, J. Mechalakos, et al. (2015). Clinical Experience with two Frameless Stereotactic Radiosurgery (fsrs) Systems using Optical Surface Imaging for Motion Monitoring. *Journal of Applied Clinical Medical Physics/American College of Medical Physics 16*(4), 5416. 12

Li, Y. and J. Lei (2010). A Feasible Solution To The Beam-angle-optimization Problem In Radiotherapy Planning With A Dna-based Genetic Algorithm. *IEEE Transactions on biomedical engineering 57*(3), 499–508. 21

Lim, G. J., M. C. Ferris, S. J. Wright, D. M. Shepard, and M. A. Earl (2007). An optimization framework for conformal radiation treatment planning. *INFORMS Journal on Computing 19*(3), 366–380. 22

Liu, X., A. H. Belcher, Z. Grelewicz, and R. D. Wiersma (2015). Robotic stage for head motion correction in stereotactic radiosurgery. In *2015 American Control Conference (ACC)*, pp. 5776–5781. IEEE. 13, 135

Ljung, L. (1999). *System Identification Theory for the User* (Second ed.). Upper Saddle River, NJ, USA.: Prentice Hall. 78

Long, T. C. (2015). Optimization problems in radiation therapy treatment planning. 118

Lu, R., R. J. Radke, L. Hong, C.-S. Chui, J. Xiong, E. Yorke, and A. Jackson (2006). Learning The Relationship Between Patient Geometry And Beam Intensity In Breast Intensity-modulated Radiotherapy. *IEEE transactions on biomedical engineering 53*(5), 908–920. 21

Merlet, J. (2015). *Parallel robots*. Springer. 26, 44

Mnih, V., K. Kavukcuoglu, D. Silver, A. a. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis (2015). Human-level Control Through Deep Reinforcement Learning. *Nature* (7540), 529–533. 23, 112

Mochiyama, H. (2005). Hyper-flexible robotic manipulators. In *IEEE International Symposium on Micro-NanoMechatronics and Human Science, 2005*, pp. 41–46. IEEE. 18

Montana, D. J. (1988). The Kinematics of Contact And Grasp. *The International Journal of Robotics Research 7*(3), 17–32. 46, 55, 57, 136, 138

Mooney, M. (1940). A theory of large elastic deformation. *Journal of applied physics 11*(9), 582–592. 26, 34, 48

Murphy, M. J. and R. S. Cox (1996). The Accuracy Of Dose Localization For An Image-guided Frameless Radiosurgery System. *Medical physics 23*(12), 2043–2049. 11

Murphy, M. J., R. Eidens, E. Vertatschitsch, and J. N. Wright (2008). The effect of transponder motion on the accuracy of the calypso electromagnetic localization system. *International Journal of Radiation Oncology* Biology* Physics 72*(1), 295–299. 12

Murray, R. M. (2017). *A Mathematical Introduction To Robotic Manipulation*. CRC press. 26

Murray, R. M. and S. Sastry (1990). Grasping and Manipulation using Multifingered Robot Hands. In *Proceedings of Symposia in Applied Mathematics*, Volume 41, pp. 329–335. 54, 59, 136

Navarro-Martin, A., J. Cacicedo, O. Leaman, I. Sancho, E. García, V. Navarro, and F. Guedea (2015). Comparative analysis of thermoplastic masks versus vacuum cushions in stereotactic body radiotherapy. *Radiation Oncology 10*(1), 176. 12

Nesme, M., F. Faure, and Y. Payan (2006). Hierarchical Multi-Resolution Finite Element Model for Soft Body Simulation. *Lecture Notes in Computer Science 4072*, 40–47. 18

Nesme, M., M. Marchal, E. Promayon, M. Chabanas, Y. Payan, and F. Faure (2005). Physically Realistic Interactive Simulation For Biological Soft Tissues. *Recent Research Developments in Biomechanics 2*, 1–22. 18

Nguyen, D., D. Thomas, M. Cao, D. OConnor, J. Lamb, and K. Sheng (2016). Computerized triplet beam orientation optimization for mri-guided co-60 radiotherapy. *Medical physics 43*(10), 5667–5675. 134

Nguyen, V.-D. (1988). Constructing force-closure grasps. *The International Journal of Robotics Research 7*(3), 3–16. 46, 60

Ogden, R. (1997). *Non-linear Elastic Deformations*. Mineola, New York: Dover Publicationbs, Inc. 18, 26, 28, 30, 48, 49, 50, 66, 67, 141

Ogunmolu, O., M. Folkerts, N. Dan, N. Gans, and S. Jiang (2018). Automating Beam Orientation Optimization for IMRT Treatment Planning: A Deep Reinforcement Learning Approach. In *60th Annual Meeting of the American Association of Physicists in Medicine, Nashville, TN*. AAPM. 24

Ogunmolu, O., M. Folkerts, D. Nguyen, N. Gans, and S. Jiang (2018). Deep BOO! Automating Beam Orientation Optimization in Intensity Modulated Radiation Therapy. In *Algorithmic Foundations of Robotics, XIII Workshop, Merida, Mexico. Published in Springer's Proceedings in Advanced Robotics (SPAR) book*. 24, 117, 121, 126, 133

Ogunmolu, O., N. Gans, S. Jiang, and X. Gu (2015, June). An Image Guided Soft Robotic Patient Positioning System for Maskless Head And Neck Cancer Radiotherapy: A Proof of Concept Study. *Medical Physics: The International Journal of Medical Physics Research and Practice 42*, 3266–3266. 23

Ogunmolu, O., N. Gans, and T. Summers (2017). Robust two player, zero-sum deep reinforcement learning. 24

Ogunmolu, O., N. Gans, and T. Summers (2018). Minimax iterative dynamic game: Application to nonlinear robot control tasks. *IEEE International Conference on Robots and Intelligent Systems, (IROS), Madrid, 2018.*. 24

Ogunmolu, O., X. Gu, S. Jiang, and N. Gans (2015a, Aug). A Real-Time Soft Robotic Patient Positioning System for Maskless Head-and-Neck Cancer Radiotherapy: An Initial Investigation. In *IEEE International Conference on Automation Science and Engineering*, Gothenburg, Sweden. 81

Ogunmolu, O., X. Gu, S. Jiang, and N. Gans (2016a, Aug). Vision-based Control of a Soft Robot for Maskless Head and Neck Cancer Radiotherapy. In *IEEE International Conference on Automation Science and Engineering*, Fort Worth, Texas. 81, 88

Ogunmolu, O., A. Kulkarni, Y. Tadesse, X. Gu, S. Jiang, and N. Gans (2017). Soft-neuroadapt: A 3-dof neuro-adaptive patient pose correction system for frameless and maskless cancer radiotherapy. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, CA*, pp. 3661–3668. IEEE. 18, 23, 70, 86

Ogunmolu, O., A. Sadeghnejad Barkousaraie, N. Dan, N. Gans, and S. Jiang (2019). Deep Learning Neural Network for Beam Orientation Optimization. In *To appear at International Conference on the use of Computers in Radiation Therapy XVI*. 24

Ogunmolu, O. P., X. Gu, S. Jiang, and N. R. Gans (2015b). A real-time, soft robotic patient positioning system for maskless head-and-neck cancer radiotherapy: An initial investigation. In *Automation Science and Engineering (CASE), 2015 IEEE International Conference on, Gothenburg, Sweden*, pp. 1539–1545. IEEE. 23, 70, 71

Ogunmolu, O. P., X. Gu, S. Jiang, and N. R. Gans (2016b). Vision-based control of a soft robot for maskless head and neck cancer radiotherapy. In *Automation Science and Engineering (CASE), 2016 IEEE International Conference on, Fort Worth, Texas*, pp. 180–187. IEEE. 23, 70

OConnor, D., V. Yu, D. Nguyen, D. Ruan, and K. Sheng (2018). Fraction-variant beam orientation optimization for non-coplanar IMRT. *Physics in Medicine and Biology 63*(4). 134

Parks, P. (1966). Liapunov Redesign of Model Reference Adaptive Control Systems. *IEEE Transactions on Automatic Control 11*(3), 362–367. 91

Patino, H. and D. Liu (2000). Neural network-based model reference adaptive control system. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 30*(1), 198–204. 93

Pugachev, A., A. Boyer, and L. Xing (2000). Beam orientation optimization in intensity-modulated radiation treatment planning. *Medical Physics 27*(6), 1238–1245. 21

Pugachev, A. and L. Xing (2001). Computer-assisted selection of coplanar beam orientations in intensity-modulated radiation therapy. *Physics in Medicine & Biology 46*(9), 2467. 21

Pugachev, A. and L. Xing (2002). Incorporating prior knowledge into beam orientaton optimization in imrt. *International Journal of Radiation Oncology* Biology* Physics 54*(5), 1565–1574. 21

Pujana-Arrese, A., A. M. Esnaola, J. Arenas, I. Retolaza, A. Martinez-Esnaola, and J. Landaluze (2007). Modelling In Modelica Of A Pneumatic Muscle: Application To Model An Experimental Set-Up. 17

Quan, E. M., X. Li, Y. Li, X. Wang, R. J. Kudchadker, J. L. Johnson, D. A. Kuban, A. K. Lee, and X. Zhang (2012). A comprehensive comparison of imrt and vmat plan quality for prostate cancer treatment. *International Journal of Radiation Oncology* Biology* Physics 83*(4), 1169–1178. 134

Renda, F., M. Giorelli, M. Calisti, M. Cianchetti, and C. Laschi (2014). Dynamic model of a multibending soft robot arm driven by cables. *IEEE Transactions on Robotics 30*(5), 1109–1122. 18, 26

Renda, F. and L. Seneviratne (2018). A Geometric and Unified Approach for Modeling Soft-Rigid Multi-body Systems with Lumped and Distributed Degrees of Freedom. *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 1567 – 1574. 18

Rivlin, R. S. and D. W. Saunders (1950). Large Elastic Deformations of Isotropic Materials. VII. Experiments on the Deformation of Rubber. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 243*(865), 251–288. 26, 28, 31

Robb, D. et al. (2013). Assessing the efficiency and consistency of daily image-guided radiation therapy in a modern radiotherapy centre. *Journal of Medical Imaging and Radiation Sciences*. 7

Roger, B. M. (1991). *Game Theory: Analysis of Conflict*. Harvard University Press. 106, 107

Romeijn, H. E., R. K. Ahuja, J. F. Dempsey, and A. Kumar (2005). A column generation approach to radiation therapy treatment planning using aperture modulation. *SIAM Journal on Optimization 15*(3), 838–862. 118

Rucker, D. C., B. A. Jones, and R. J. Webster III (2010). A geometrically exact model for externally loaded concentric-tube continuum robots. *IEEE transactions on robotics: a publication of the IEEE Robotics and Automation Society 26*(5), 769. 18

Rus, Daniela; Tolley, Michael T. (2015). Design, Fabrication And Control Of Soft Robots. *Nature*, 467–475. 18

Rusu, R. B. (2009). Semantic 3D object Maps for Everyday Manipulation in Human Living Environments. *PhD thesis*. 84, 86

Rusu, R. B., Z. C. Marton, N. Blodow, M. E. Dolha, and M. Beetz (2008). Functional Object Mapping of Kitchen Environments. *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, 3525–3532. 85

Sadeghnejad Barkousaraie, A., O. Ogunmolu, S. Jiang, and D. Nguyen (2019a). A Fast Deep Learning Approach for Beam Orientation Selection Using Supervised Learning with Column Generation on IMRT Prostate Cancer Patients. *Medical Physics, American Association of Physicists in Medicine*. 24, 118, 119, 129, 130, 132

Sadeghnejad Barkousaraie, A., O. Ogunmolu, S. Jiang, and D. Nguyen (2019b). A Reinforcement Learning Application of Guided Monte Carlo Tree Search Algorithm for Beam Orientation Selection in Radiation Therapy. *Medical physics. 46*(6), E237–E237. 24

Sadeghnejad Barkousaraie, A., O. Ogunmolu, D. Nguyen, and S. Jiang (2019). Deep Learning Neural Network for Beam Orientation Optimization. In *To appear at International Conference on the use of Computers in Radiation Therapy XVI*. 24, 117, 130, 131, 133

Schadd, M. P., M. H. Winands, H. J. Van Den Herik, G. M.-B. Chaslot, and J. W. Uiterwijk (2008). Single-player monte-carlo tree search. In *International Conference on Computers and Games*, pp. 1–12. Springer. 123

Schweikard, A., R. Tombropoulos, and J. R. Adler (1995). Robotic Radiosurgery with Beams of Adaptable Shapes. In N. Ayache (Ed.), *Computer Vision, Virtual Reality and Robotics in Medicine*, Berlin, Heidelberg, pp. 138–149. Springer Berlin Heidelberg. 7

Sedal, A., D. Bruder, J. Bishop-Moser, R. Vasudevan, and S. Kota (2018). A continuum model for fiber-reinforced soft robot actuators. *Journal of Mechanisms and Robotics 10*(2), 024501. 18, 25

Shepherd, R. F., F. Ilievski, W. Choi, S. A. Morin, A. A. Stokes, A. D. Mazzeo, X. Chen, M. Wang, and G. M. Whitesides (2011). Multigait soft robot. *Proceedings of the National Academy of Sciences 108*(51), 20400–20403. 17

Silver, D., A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis (2016). Mastering the game of Go with deep neural networks and tree search. *nature 529, no. 7587: 484-489.*. 23, 98

Silver, D., J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, et al. (2017). Mastering the Game Of Go Without Human Knowledge. *Nature 550*(7676), 354. 23, 98, 123

Södertröm, S. and A. Brahme (1993). Optimization of the Dose Delivery In A Few Field Techniques Using Radiobiological Objective Functions. *Medical physics 20*(4), 1201–1210. 20

Spivak, M. (1979). A Comprehensive Introduction to Differential Geometry. Vol. V. Berkeley: Publish or Perish. *Inc. XI*. 53

Stein, J., R. Mohan, X.-H. Wang, T. Bortfeld, Q. Wu, K. Preiser, C. C. Ling, and W. Schlegel (1997). Number and orientations of beams in intensity-modulated radiation treatments. *Medical Physics 24*(2), 149–160. 20, 21

Sterzing, F., R. Engenhart-Cabillic, M. Flentje, and J. Debus (2011). Image-guided radiotherapy: a new dimension in radiation oncology. *Deutsches Aerzteblatt International 108*(16), 274. 23

Summers, T., O. Ogunmolu, N. Gans, V. Renganathan, N. Kariotoglou, M. Kamgarpour, et al. (2017). Robustness margins and robust guided policy search for deep reinforcement learning. In *IEEE/RSJ International Conference on Robots and Intelligent Systems,(Abstract Only Track)*, Volume 8. 24

Takakura, T. et al. (2010). *The geometric accuracy of frameless stereotactic radiosurgery using a 6D robotic couch system*. Phys Med Biol. 8

Takakura, T., T. Mizowaki, M. Nakata, S. Yano, T. Fujimoto, Y. Miyabe, M. Nakamura, and M. Hiraoka (2009). The geometric accuracy of frameless stereotactic radiosurgery using a 6d robotic couch system. *Physics in Medicine & Biology 55*(1), 1. 2, 8

Torr, Philip HS and Zisserman, A. (2000). MLESAC: A New Robust Estimator with Application to Estimating Image Geometry. *Computer Vision and Image Understanding 78*(1), 138–156. 84

Treloar, L. R. G. (1975). *The physics of rubber elasticity*. Oxford University Press, USA. 26, 28, 31, 34

Trivedi, D., A. Lotfi, and C. D. Rahn (2008). Geometrically Exact Models For Soft Robotic Manipulators. *IEEE Transactions on Robotics 24*(4), 773–780. 26

Trivedi, D., C. D. Rahn, M. Kier, and I. D. Walker (2008). Soft Robotics : Biological Inspiration , State Of The Art , And Future Research. *5*(3), 99–117. 18

Truesdell, C. and W. Noll (1965). *The Non-Linear Field Theories of Mechanics*. Springer. 35, 65

Verellen, D., M. De Ridder, and G. Storme (2008). A (short) history of image-guided radiotherapy. *Radiotherapy and Oncology 86*(1), 4–13. 7

Viola, P. and M. Jones (2001). Rapid Object Detection using a Boosted Cascade of Simple Features. 73

Wang, C., J. Dai, and Y. Hu (2003). Optimization Of Beam Orientations And Beam Weights For Conformal Radiotherapy Using Mixed Integer Programming. *Physics in Medicine & Biology 48*(24), 4065. 22

Wang, J. S. and Y. P. Chen (2006). A Fully Automated Recurrent Neural Network for Unknown Dynamic System Identification and Control. *IEEE Transactions on Circuits and Systems 53*. 93

Webb, S. (1999). Conformal intensity-modulated radiotherapy (imrt) delivered by robotic linac-testing imrt to the limit? *Physics in Medicine & Biology 44*(7), 1639. 7

Webb, S. (2000). Conformal intensity-modulated radiotherapy (imrt) delivered by robotic linac-conformality versus efficiency of dose delivery. *Physics in Medicine and Biology 45*(7), 1715. 7

Webb, S. (2001). *Intensity-Modulated Radiation Therapy*. Institute of Physics Publishing Ltd, Bristol and Philadelphia. x, 4, 5, 6, 7, 19

Werbos, P. J. (1990). Backpropagation Through Time: What It Does and How to Do It. *Proceedings of the IEEE 78*(10), 1550–1560. 94

Wiersma, R. D., Z. Wen, M. Sadinski, K. Farrey, and K. M. Yenice (2009). Development of a frameless stereotactic radiosurgery system based on real-time 6d position monitoring and adaptive head motion compensation. *Physics in Medicine & Biology 55*(2), 389. 135

Woods, K., D. Nguyen, A. Tran, Y. Y. Victoria, M. Cao, T. Niu, P. Lee, and K. Sheng (2016). Viability of noncoplanar vmat for liver sbrt compared with coplanar vmat and beam orientation optimized 4π imrt. *Advances in radiation oncology 1*(1), 67–75. 129, 134

Xia, Y. and G. M. Whitesides (1998). Soft lithography. *Angewandte Chemie International Edition 37*(5), 550–575. 17

Xing, L. (2000a). Dosimetric effects of patient displacement and collimator and gantry angle misalignment on intensity modulated radiation therapy. *Radiother Oncol 56*(1), 97–108. 2, 8

Xing, L. (2000b). Dosimetric effects of patient displacement and collimator and gantry angle misalignment on intensity modulated radiation therapy. *Radiother Oncol 56*(1), 97–108. 7

Yekutieli, Y., R. Sagiv-Zohar, R. Aharonov, Y. Engel, B. Hochner, and T. Flash (2005). A dynamic model of the octopus arm. i. biomechanics of the octopus reaching movement. *Journal of neurophysiology*. 18

Zambrano, D., M. Cianchetti, C. Laschi, H. Hauser, R. Füchslin, and R. Pfeifer (2014). The Morphological Computation Principles As A New Paradigm For Robotic Design. *Opinions and Outlooks on Morphological Computation*, 214–225. 16, 25

Zheng, T., D. T. Branson, R. Kang, M. Cianchetti, E. Guglielmino, M. Follador, G. A. Medrano-Cerda, I. S. Godage, and D. G. Caldwell (2012). Dynamic continuum arm model for use with underwater robotic manipulators inspired by octopus vulgaris. In *2012 IEEE International Conference on Robotics and Automation*, pp. 5289–5294. IEEE. 18

BIOGRAPHICAL SKETCH

Olalekan Ogunmolu is currently a PhD candidate in Electrical Engineering at The University of Texas at Dallas. He obtained his master'ss degree in control systems from The University of Sheffield in 2012. In addition to being part of the SeRViCe Lab, he is also a member of the Medical Artificial Intelligence and Automation Lab at UT Southwestern's Radiation Oncology department.

During his PhD, he was a robotics engineering intern at Amazon Robotics where he developed the vision segmentation software for line scanners used at the Newark Fulfillment Center. He also worked on the Hercules mobile robot project. In 2018, he was a robotics research intern at Preferred Networks in Tokyo, Japan. There, he developed a stable Lyapunov solver using a Gaussian mixture model for robot motion planning.

He is broadly interested in the applications of finite elastic deformation, approximate dynamic programming (reinforcement learning), and control theory to bleeding-edge soft and rigid robot motion tasks.

CURRICULUM VITAE

Olalekan P. Ogunmolu

June 3, 2019

Contact Information:

Department of Electrical Engineering
The University of Texas at Dallas
800 W. Campbell Rd.
Richardson, TX 75080-3021, U.S.A.

Voice: N/A
Fax: N/A
Email: opo140030@utdallas.edu

Educational History:

M.Sc in Eng. in Control Systems, The University of Sheffield, 2013
Ph.D., Electrical Engineering, The University of Texas at Dallas, 2019

A Multi-DOF Soft Robot Mechanism for Patient Motion Correction and Beam Orientation Selection in Cancer Radiation Therapy
Ph.D. Dissertation
Electrical Engineering Department, University of Texas at Dallas
Advisors: Drs. Nick Gans (UTD) and Steve B. Jiang (UTSW)

Autonomous Navigation of a Rotorcraft Unmanned Vehicle Using Machine Vision
MSc Thesis
Autonomous Control and Systems Engineering Department, The University of Sheffield
Advisor: Professor Tony J. Dodd

Employment History:

Postdoctoral Scholar, The University of Pennsylvania, Perelman School of Medicine, June 2019 – Present