TRAJECTORY SOLUTIONS FOR A GAME-PLAYING ROBOT USING NONPREHENSILE MANIPULATION METHODS AND MACHINE VISION

by

Senthan Mathavan

A thesis submitted in partial fulfilment of the requirement for the award of

Doctor of Philosophy

Loughborough University

August 2009

ABSTRACT

The need for autonomous systems designed to play games, both strategy-based and physical, comes from the quest to model human behaviour under tough and competitive environments that require human skill at its best. In the last two decades, and especially after the 1996 defeat of the world chess champion by a chess-playing computer, physical games have been receiving greater attention. Robocup^{TM}, i.e. robotic football, is a well-known example, with the participation of thousands of researchers all over the world. The robots created to play snooker/pool/billiards are placed in this context. Snooker, as well as being a game of strategy, also requires accurate physical manipulation skills from the player, and these two aspects qualify snooker as a potential game for autonomous system development research. Although research into playing strategy in snooker has made considerable progress using various artificial intelligence methods, the physical manipulation part of the game is not fully addressed by the robots created so far. This thesis looks at the different ball manipulation options snooker players use, like the shots that impart spin to the ball in order to accurately position the balls on the table, by trying to predict the ball trajectories under the action of various dynamic phenomena, such as impacts.

A 3-degree of freedom robot, which can manipulate the snooker cue on a par with humans, at high velocities, using a servomotor, and position the snooker cue on the ball accurately with the help of a stepper drive, is designed and fabricated. Using a single, stationary, overhead camera and image processing techniques, the balls' movements on the snooker table are tracked to 1mm spatial accuracy. The tracking results are used to determine various parameters, like friction coefficients and coefficients of restitution, involved in the ball dynamics. Some efforts on determining ball spin, over a limited area, by tracking a circular pattern put on the ball's surface, are also presented. A thin-film force sensor has also been installed in the snooker cue, close to its tip, and the force measurements are used in conjunction with the camera tracking to arrive at some conclusions regarding the cueing dynamics. Moreover, the friction during the collision between two snooker balls and that between a ball and cushion are theoretically analysed using the principles of impact mechanics. Contrary to previous works, no constraints are placed on the direction of slip between bodies during impact. Differential equations describing the ball motion during impact are

obtained, and then solved by the use of numerical algorithms to obtain solutions for the post-impact ball trajectories. Finally, the problem of ball positioning in snooker is introduced. The close relationship between the problem and a specialised robotic manipulation domain called nonprehensile manipulation, which is concerned with the positioning of objects without grasping them, is established. An artificial neural network-based model is developed for the dynamic interaction between the cue and the cue ball during cueing. A forward dynamics model for the ball motion is put forward by combining the results obtained by the camera-based tracking and the numerical and empirical analyses of various collisions, as mentioned earlier. Given the desired final ball locations on the table, optimisation is chosen as the proposed solution for the ball positioning by minimising the error distances on the table in order to obtain the required control parameters of the robot. Genetic Algorithms are used for this derivative-free optimisation. Experiments performed using the robot, in a 5 ft x 6ft area of the table, indicate that the optimal solution for the robot parameters are able to position the cue ball to an accuracy of 100-250mm and give an object ball potting accuracy of over 90%.

Keywords: game-playing robots, snooker robots, billiards robots, non-prehensile manipulation, impulsive manipulation, ball-tracking, spin-tracking, ball collision mechanics, ball trajectories, trajectory optimisation.

ACKNOWLEDGEMENTS

The research work reported in this thesis was carried out at the Mechatronics Research Centre, Department of Mechanical and Manufacturing Engineering, at Loughborough University, during the period 2006-2009. This work was supported by an Overseas Research Student Award and a departmental scholarship.

I would like to express my deepest gratitude to Professor Mike Jackson and Professor Rob Parkin for their guidance, support and encouragement throughout this work. Your insightful suggestions during the numerous discussions that we had were invaluable for this research. I am also indebted to Professor Mike Jackson for his helpful comments on the final version of the thesis.

I am grateful for the camaraderie shown by the colleagues at the Holywell Mechatronics Research Centre from day one. Many thanks for all your help, both technical and in general, advice and suggestions that have been immensely useful.

Many thanks to David Liddell for the several technical discussions that we had and also for his help in sorting out some issues with the stepper drive.

Bob Ludlam and Robert Smith of the Engineering Workshop had been more than helpful during the manufacturing stage of the components and deserve my sincere thanks.

Thanks to Jo Mason for helping me out with the paperwork.

The security staff of the university had been nice enough in tolerating my nocturnal working hours in the second half of my research and always had some pleasant words to say.

Special thanks are also due to friends from my undergrad years for their care and concern.

TABLE OF CONTENTS

LIST OF SYMBOLS

α	angle defining the cue hitting point on the cue ball/ amount of cue *squirt*	rad
β	fractional impact parameter for oblique collisions	-
a	acceleration	m/s^2
b	object ball offset distance in an oblique collision	m
δ	swing of the forearm during cueing	rad
Δ	shot difficulty index / a small increment in of a parameter	-
Δt	Duration of impulse	s
d	distances between balls and pockets on the table	m
e	energetic coefficient of restitution between balls	-
e	energetic coefficient of restitution between ball and cushion	-
E	energy	J
$\dot{E}$	power	W
φ	swing of the wrist during cueing	rad
Φ	direction of a body slip during impulse	rad
F	force	N
γ	included angle between balls on the table plane	rad
g	gravitational acceleration	mm/s^2
I	impulse	Ns
μ_{bb}	sliding friction coefficient between balls	-
μ_r	coefficient of rolling friction	-
μ_s	coefficient of sliding friction	-
μ_w	sliding friction coefficient between ball and cushion	-
M	ball mass	kg
N	number of iterations in numerical scheme	-
ω	angular velocity	rad/s
ω^S	ball sidespin	rad/s
ω^T	ball topspin	rad/s
P	impulse	N.s
q	state vector consisting of robot parameters	-
ρ	horizontal shift in the reaction force on the ball from the table	m

r	radius of cue tip	m
R	ball radius	m
$\boldsymbol{R_c}$	rotation matrix between camera frame and the world coordinates	-
ψ	included angle between a cushion and a ball trajectory/ angular shift of axis system for massé shots	rad
s,s'	slip speed during impulse	m/s
θ	direction of cue ball movement/ offset angle for oblique impact	rad
θ_C	swivel angle of the cue on the table	rad
t	time	s
$\boldsymbol{Tc}$	translation matrix between camera frame and the world coordinates	-
u	velocity	m/s
V	velocity	m/s
V_0	initial cue ball velocity	m/s
V_{C0}	cue velocity at impact with cue ball	m/s
W	work done	J
x'_0	horizontal movement of stepper platform with respect to stun point	m
y'_0	vertical movement of stepper platform with respect to stun point	m

LIST OF ABBREVIATIONS

ANN	Artificial Neural Network
COG	Centre Of Gravity
DOF	Degrees Of Freedom
fps	Frames Per Second
LWL	Locally Weighted Learning
RMS	Root Mean Square
ROI	Region Of Interest

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

1.1 Background

1.1.1 Why Game-Playing Robots?

Games have always been of keen interest for the science community; numerous examples can be quoted. One early work that is also of particular interest to this thesis, written in the 18th century by the famous French scientist Coriolis, is a detailed treatise about the physics involved with billiards **[Nadler 2005]**. Thus it is not surprising that, when engineering research started focussing on creating autonomous systems, games became an obvious subject for study. In fact, developing strategies for board games (generally referred to as 'game playing') was one of the very first tasks undertaken in Artificial Intelligence (AI) **[Russell and Norvig 2005]**.

Games present very tough, competitive environments; hence, the intended autonomous game-playing systems have to perform well under such difficult circumstances, and this aspect sets a truly challenging research goal. Moreover, the games also present researchers with a real-world environment to tackle. Thus, the solutions developed to overcome the problems in game playing can possibly be translated directly into the other environments that humans deal with, for example an industrial situation. What is more, the robots/systems created to play games currently receive wide media attention. They have thereby become a channel through which the research community interacts with the general public.

Games such as chess **[Campbell *et al.* 2002]**, checkers **[Fogel and Chellapilla 2002]**, backgammon, Go, Othello, poker and bridge have been researched since the inception of AI in the early 1950s. Most of these are board games that purely challenge the human intellect. Thus, the line of research on these games is about strategy, planning and creating novel techniques that bring forth better game engines. The game

environment is essentially the software where the state of the game at any given time is stored. Most importantly, in the case of strategy games, the artificial system need not have any physical elements. Four decades of research on chess and on AI in general culminated in the victory of IBM's 'Deep Blue' over the World Chess Champion, Garry Kasparov in 1997.

In the last two decades, the focus has been shifting more toward the games that require a physical engagement with an opponent, the main reason being the availability of increasingly reliable and cheap hardware solutions. The ever-improving hardware capabilities of the sensors, actuators and computers along with the arrival of soft computing paradigms, such as neuro-fuzzy modelling, that perform better under uncertainties, have made the research into the physical game domains more feasible. All these factors have lead to the development of a number of well-performing, game-playing robots.

The game-playing robots have given rich insights into disciplines such as computer vision, controls, machine intelligence and machine learning which have lead to the development of a whole range of new theories and technologies. Currently, RoboCup™ (The Robot World Cup Initiative) gets extensive attention, with hundreds of researchers participating from all over the world; RoboCup™ has currently chosen soccer as its standard task **[Asada *et al.* 1999]**. To cite an example, Candea *et al.* **[2001]** state that RoboCup™ is spearheading the research on cooperative-agent modelling, a general AI research topic. Research into game-playing systems develops solutions that transcend the domain of a particular game and thus makes the development of game playing systems very valuable.

1.1.2 A Brief Overview of Game Playing Robots/Systems

In the mid-late 1980s, a robot was developed in the AT&T Bell Labs to play table tennis (ping-pong) **[Andersson 1989]**. The system was equipped with two cameras to track the 3-D motion of the ball and a racket that was connected to a robot arm through an elongated rod. From 1985-1988, a system called the Snooker Machine, a robotic system to play snooker, was built at Bristol University, UK, and, according to

Ho *et al.* **[2007]**, this had also received BBC televised coverage by 1988. However, other sports domains did not receive any attention in this regard in the 80's or, at least, there were no major developments reported during this time.

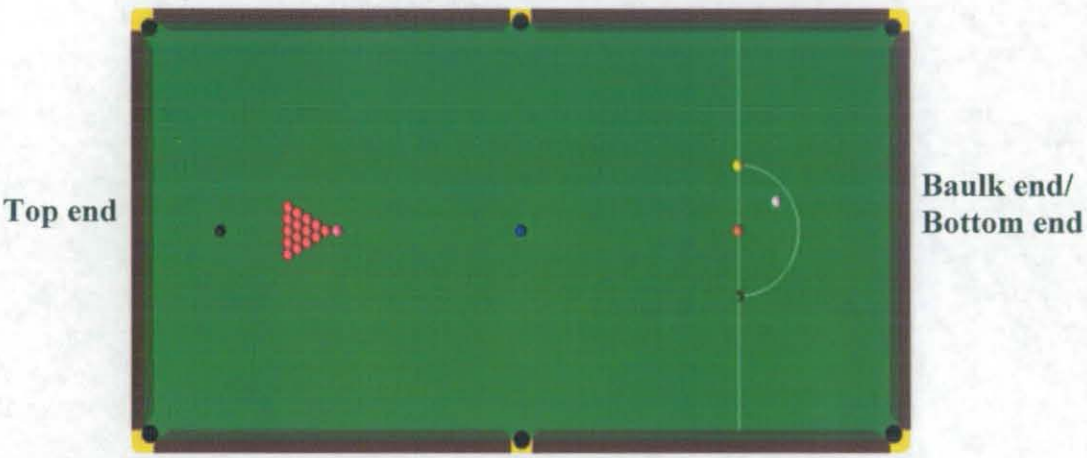
In the early 1990s, robotic football research started to gain momentum following a paper from Prof. Alan Mackworth of the University of British Columbia in 1993, where the scope of this research was pointed out for the first time **[Asada *et al.* 1999]**. Around the same time, research papers on RoboCup™ and on robots to play other games, started to appear. Since then, the research and development of game-playing robotic systems appears to be on the increase. Numerous publications on gaming robots are coming out each year, and a few dedicated journals/conferences in this domain have arrived due to the increased volume of publications. For example, the IEEE Symposium on Computational Intelligence and Games is a conference where the researchers on physical/strategy-based game playing systems (except robotic football) present their research annually.

A list of the current systems is exhaustive. There appears to be some form of sports-robot-related research carried out within every research group on robotics/mechatronics in most universities. Some examples of the systems created to play games, apart from RoboCup™, are, table tennis **[Matsushima *et al.* 2005]**, air-hockey **[Park 2001]**, yo-yo **[Jin and Zacksenhouse 2004, Hashimoto and Noritsugu 1996]**, golf **[Ming and Kajitani 2003]**, bounce juggling **[Ronsse *et al.* 2006]** and batting **[Senoo *et al.* 2004]**.

1.1.3 The Game of Snooker: An Overview

Snooker comes under the category of games collectively known as billiards that include pool (there are variations like 8-ball and 9-ball) as well as carrom. These games are collectively known as cue sports, where a stick (called a cue) is used to manipulate spherical balls on a table with pockets. The player uses the cue stick to strike a designated ball, called the cue ball, setting it in motion (this process is known as cueing). The cue ball is struck in such a manner as to collide with the other balls on the table (called the objects balls) either to pocket them in one of the six pockets or to

position them at a desired spot on the table. The objective of the game is to pocket all the object balls, except the cue ball, in a specific order. Here, it is also interesting to note that snooker is a game traditionally used in physics to illustrate the frictional rolling motion of spherical particles on a flat surface and the collisions between spherical bodies.



White- Cue ball, Reds- 1 point each, Yellow-2 points, Green- 3, Brown- 4, Blue- 5, Pink- 6, Black- 7.

Fig. 1.1. Snooker table with initial ball positions - top view

At the start of the game of snooker, the balls are arranged as shown in Figure 1.1. The cue ball is placed within the half circle at the baulk end and a shot is taken (called the break shot). A player has to pot a red ball followed by any other colour ball, the cue ball is played from wherever it ends up after a shot, and the player can continue to do so until a ball fails to be potted, and the points are awarded as given in Figure 1.1. The potted colour ball is put back on the table in its designated initial position as shown in Figure 1.1. When a player fails to pocket a ball the turn passes on to the other player. When all the reds on the table have been pocketed the players then start to pot the colour balls in the order of increasing value. For a given arrangement of the balls as shown above, a frame is said to be complete when the black ball is potted and the player that has the most points in that frame wins the frame. A match can have a number of frames, and the winner is the one who wins the most number of frames. For a detailed description of the rules, refer to the official website of the World Professional Billiards & Snooker Association (WPBSA) at www.worldsnooker.com.

Snooker is fast becoming a popular sport, and globally there has been a resurgence of interest in this game in the last decade with the game reaching the Far East.

> "In the Far East, the snooker boom is developing at pace. The China Open staged in Beijing in 2005 was enthusiastically received by Chinese fans. A staggering 110 million people watched live on television as Ding Junhui beat Stephen Hendry in the final to elevate himself to the status of national hero. Last year, in 2007, the BBC generated 138 hours of snooker coverage, reaching over 27 million viewers in the UK (45% of the population), while Eurosport® provided 144 hours, reaching 57 million people" **[WPBSA-World Professional Billiards & Snooker Association, from www.worldsnooker.com, accessed on 28.05.2008]**.

The WPBSA also says that in the UK, snooker is played by 4 million adults. There are more than 800 snooker clubs in the UK with an average of 20 tables per club, according to a database of snooker clubs **[Click-Snooker, 2008]**. There are also hundreds of snooker leagues held throughout the UK every year, with thousands of participants. A slight variation of this game, called pool, is more popular in North America, but the fundamentals are very similar. All this indicates that snooker is a sport that enjoys a very good reception from the general public.

1.1.4 What is the Scope of a Robotic System for Snooker?

50 years of AI research on game strategy resulted in creating a system that could beat the World Chess Champion. When it comes to physical games, according to Asada *et al.* **[1999]**, RoboCup™ has set itself the following goal:

> "By the mid-21st century (2050), a team of autonomous humanoid robots shall beat the human World Cup champion team under the official regulations of FIFA."

When compared to humanoid soccer, snooker and billiards presents a relatively lesser physical challenge. By the very nature of the game, snooker does not demand a very fast robotic response, rather it requires a very accurate one. Hence, the computer

vision, decision-making, and robotic elements of the system do not need to operate in real-time. In addition to this, snooker only has a single agent actively present at any one time in the game environment, whereas in soccer multiple agents are present and this complicates the decision-making part of the system. Hence, a snooker/billiards robot has a good potential to be the first robotic system to beat a human in a standard physical game, and this can probably be achieved well before the target of 2050 set by RoboCup™. When the overview of the robotic configurations (especially the kinematic solutions of some of the robots) is explained later in this chapter, this argument becomes clearer. This aspect of snooker presents a very exciting prospect.

Earlier, it was seen that billiards/snooker has a very popular appeal, which is on the rise. Therefore it is reasonable to assume that a large number of amateurs will embark on this sport every year (even an approximate figure was nowhere to be found). Currently, snooker training does not involve any technology whatsoever, and the coaching is carried out by professional trainers. In this context, the development of the robotic system elicits a proper understanding of the science behind the human skills and dynamics involved with this game. As the fundamental technology is understood and developed, a possible spin-off may be a multi-media-based dedicated training system for snooker that trains amateur players from the very start. Elements involving multimedia technology can also be developed to assist the current training processes in a better way to enhance the training experience.

1.1.5 Robotic-Systems for Playing Snooker/Pool/Billiards

This section gives a general introduction to the systems that currently play the billiard family games.

1.1.5.1 Bristol University, UK

In the mid-late 90s, Bristol University developed a robotic snooker player in its Advanced Manufacturing and Automation Research Centre. "The main objective was to demonstrate the feasibility of automating a complex task that demands a high level of human skill and decision making" **[Shu 1994, Ho *et al.* 2007]**. They came up with a solution of using a PUMA 560 manipulator arm, which holds a cue launcher that in

turn strikes the cue ball. The manipulator arm is attached to an overhead gantry robot (a SKF Linear Drive System) that moves the manipulator arm over the whole table area. The cue is powered by a pressure-regulated pneumatic cylinder. An overhead camera is used to roughly locate the balls on the table and its images are used to select an appropriate strategy. An additional cue-mounted camera is used for visual servoing to accurately position the cue on the ball. The project documentation (a PhD thesis) addresses the automation aspects of such a system. In addition, considerable attention has been given to the development of an appropriate playing strategy for the robot. However, the system's performance, like the most critical ball-potting accuracy, is scarcely documented, apart from that for some special kinds of shots, like straight-line shots and angled shots with specific predetermined angles **[Shu 1994]**. The robot treats the various impacts that take place in snooker by using very primitive models.

1.1.5.2 Sharif University, Iran

The project makes use of an overhead XYZ prismatic gantry with a revolute end-effector carrying a pneumatically powered cue to play pool **[Alian *et al.* 2004]**. The robot selects its best shot by fuzzy-based reasoning.

1.1.5.3 Tamkang University, Taiwan

A 5-degrees of freedom billiard robot based on an XYZ gantry is fixed to the table. The system uses an overhead CCD camera to image the table state. However, the way the system has been built, it is not possible for humans to play on it, as the heavy structure of the robot is rigidly attached to the sides of the table **[Cheng *et al.* 2004]**.

1.1.5.4 Adelaide University, Australia

A pool-playing system with a gantry arrangement using a solenoid-actuated ball launching system is mounted on the pool table itself **[Medwell *et al.* 2004]**. The system was developed as part of an undergraduate project.

1.1.5.5 MIT, USA

A robot was created for an undergraduate project. Two research papers, whose main focus has been on testing a certain machine-learning algorithm on several different systems, are found to have some descriptions of this robot **[Moore 1991, Moore *et al.* 1995]**. The robot uses a motorised cue and yaw control. The researchers who

implemented the machine-learning algorithms have used the ball tracking to extract its trajectory for the learning algorithm. It is a very minimal robot, and is fixed by the side of the pool table.

1.1.5.6 Queen's University, Canada

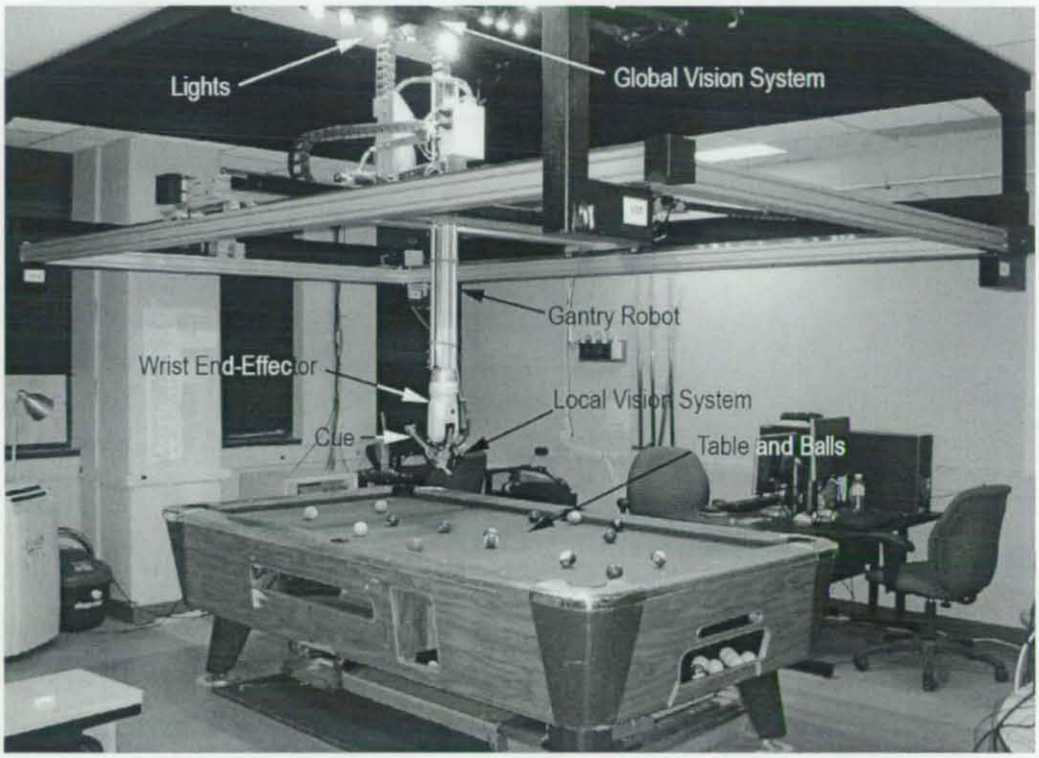


Fig. 1.2. Gantry-based pool playing system **[Greenspan *et al.* 2008]**

By far the most widely reported system in terms of general publicity (it was also featured on Discovery Channel, etc), and in terms of research output, measured by the number of research papers **[Long *et al.* 2004]**, is a system funded by the Science and Engineering Research Council of Canada and named 'Deep Green'. It again uses an industrial gantry from which a robot manipulator carrying the cue manipulation element is suspended (see Figures 1.2, 1.3). A custom-designed linear actuator is used for the cue manipulation. An 8 ft x 4 ft pool table is used for the project. The robot is reported to have a spatial positioning accuracy of 0.6 mm and an accuracy of 67% is claimed for the straight shots **[Long *et al.* 2004]**. It uses both an overhead camera and an eye-on-hand camera (for local imaging). Lam *et al.* **[2006]** had only given the angular deviation in the moving direction of the cue ball as a measure of the robot accuracy. In their latest publication, Greenspan *et al.* **[2008]** state that the robot has

pocketed runs of four consecutive balls. However, the overall performance of the robot in terms of its potting accuracy has not been reported in their latest work.

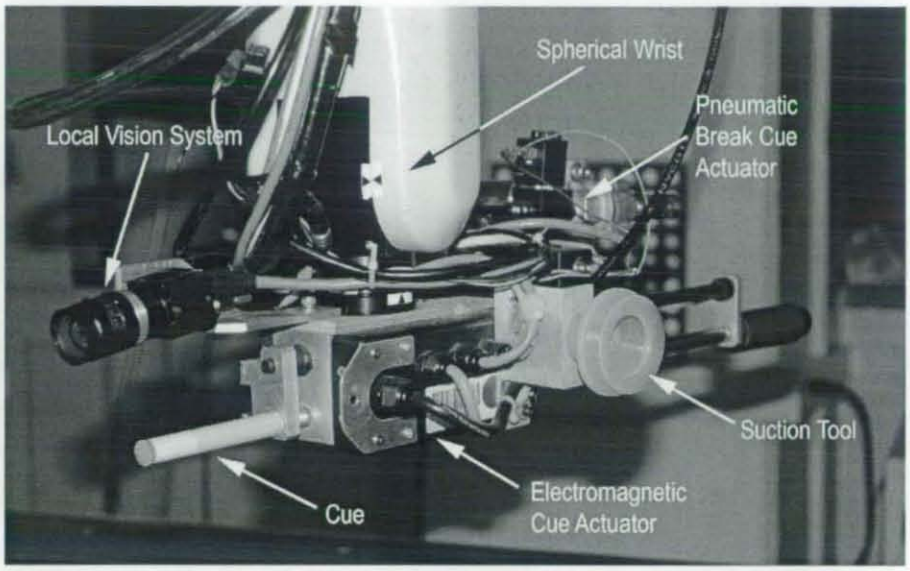


Fig. 1.3. The end effector of the Queen's University robot **[Greenspan *et al.* 2008]**

1.1.5.7 University of Alberta, Canada

PickPocket, an artificial intelligence software for computer billiards, was created during research undertaken as part of a Master's degree **[Smith 2006a]**. This is, by far, the best strategy program developed for the machine-based decision-making for billiards. PickPocket was the winner of the simulated 8-ball pool tournaments at the 10th and 11th Computer Olympiad competitions in 2003 and 2005 respectively.

A few other research projects have been focussing on producing systems that can train amateur pool/snooker players. Jebara *et al.* **[1997]** describe a wearable computer that guides the players to select the easiest shot. Larsen *et al.* **[2002]** report a system that interacts with players using audio-visual / graphics media. Since these systems address some of the issues related to computer vision and strategy in snooker, they are also reviewed in Chapter 2.

Today's mammoth computer gaming industry continues to simulate many of the physical games in a virtual environment. A number of programs to play

snooker/billiards/pool have been produced, for example, Snooker Simulation **[Grogono 2005]**. The computer games have the game strategy element in common with the robotic systems, although the former deal with a rather more idealised environment. Hence, proper attention should be given to these game engines when appropriate strategies are formulated for a snooker robot.

1.1.6 What has been achieved so far?

A careful review of the documentations of the developed systems, also the publications, leads to the following conclusions. A complete kinematic solution has been achieved for the problem of robotic billiards, in the form of gantry-based robots (however alternate configurations are still possible). The systems described by Greenspan *et al.* **[2008]** and Ho *et al.* **[2007]** have access to most of the table area, thus they are capable of executing almost all possible shots that humans play on the table. Machine vision elements were perfected in most of the systems and all of them sense the static ball positions. Considerable achievements have been made in the strategy formulation for the best-shot selection and in the search for potential future shots, thus creating a shot sequence, at the planning level.

1.2 Motivation

The scope of a snooker robot, as described in Section 1.4.1, combined with the deficiencies of the current systems, as described below, provides the motivation for this project. The literature review and an analogy of the human snooker-playing techniques, combined with the comparison of other types of robotic systems in general, and game playing robotic systems in particular, enable us to identify the flaws/deficiencies of the current systems. An analogy of the human skills involved in this game also highlights several necessary improvements that need to be developed to succeed in playing this game up to the standards of humans.

1.2.1 Deficiencies with the Current Systems

Many issues related to the human skills involved with snooker have not been addressed properly by the above-described systems. Some inherent aspects of the game have been left out because an ill-conceived system design could not tackle them. Inadequate attention has been paid to the internal system dynamics (the physics involved in the game) that play a major part in the success of a system of this kind. Alternate methodologies, apart from the traditional programming-based approach for shot-making, have not been explored properly. The following are found to be the major problems with the existing systems.

1.2.1.1 Understanding of the System Dynamics

All of the systems mentioned hitherto are supposed to use a programming-based solution to the problem of manipulation. Such solutions usually require a system model, based on its dynamics. However, the literature on system dynamics scarcely exists for snooker and billiards in general. Some early experimentation has been performed by researchers on billiards physics, but the comprehensive experimentation that is necessary to describe the ball-table dynamics is needed. Also, more accurate impacts models are needed to predict the ball trajectories accurately.

1.2.1.2 Disregard for some Salient Aspects of the Game

Snooker players achieve a great deal of variety in their game by introducing spin onto the ball by striking the ball at different points on its surface (see Chapter 2 for details). Not a single reported effort appears to exist on this very important facet of the game.

1.2.1.3 Trajectory Solutions

Snooker is all about controlling the ball trajectories accurately as planned. Suitable manipulation algorithms/schemes are needed to ensure that the decisions made by the strategy element are executed accordingly. This aspect has not been addressed in the literature.

1.2.1.4 Alternate Methodologies for Shot Planning

As a replacement for the traditional programming-based approach, alternate methods like machine learning can also be used. These methods have been successfully used in

other games where the system dynamics modelling was not completely possible, e.g. table tennis **[Matsushima *et al.* 2003, 2005]**. The machine-learning-based models, if used, will eliminate the need to consider the issues that are raised in Sections 1.2.1.1 and 1.2.1.3.

1.3 Research Novelty

This research work claims the usage of the following novel techniques and methodologies:

- A robotic system that uses in its development a novel way of manipulation that closely resembles human cueing e.g., efforts to impart different types of spin on to the ball by the way of precise positioning of the cue on the cue ball.
- Snooker cue instrumented with a force sensor and its testing.
- Novel use of vision as a means of object tracking and its use to obtain the dynamic parameters involved in the ball motion.
- Some preliminary efforts/ideas about the spin tracking of a snooker ball using a single overhead camera.
- The first complete theoretical models to describe the collisions between two snooker balls, also taking into consideration various frictional interactions under such conditions. A similar process has also been used to describe the collision of the ball with a cushion. Numerical solutions have been used for the first time for snooker impacts.
- For the first time, non-prehensile manipulation methodologies have been studied from the perspective of controlling the balls so that they proceed to predetermined locations have been presented in this thesis. A Genetic Algorithm-based optimisation procedure is used for this purpose.
- When a dynamic model is not available for a dynamic interaction, a machine-learning- based approach is taken. One such method using a neural network is used for the cue-cue ball interaction.

1.5 Thesis Outline

This dissertation continues with a more elaborate treatment of the material covered in this introduction. The relevant background works are reviewed in Chapter 2. Chapter 3 outlines the research methodology and presents the overview of the proposed system. Methods used for the design and development are explained in Chapter 4. Experiments performed with a high-speed camera and a force-sensor-instrumented cue are described in Chapter 5. Chapter 6 deals with the development of theoretical models for two of the collisions encountered in snooker. Chapter 7 presents the overall solutions for ball trajectories and results. Chapter 8 discusses the present research from a critical perspective. The conclusions of the study and the future research directions are given in the final chapter.

Chapter 2

LITERATURE REVIEW

An autonomous snooker system will consist of a number of subsystems. Its makeup must include both sensors and actuators, and the software that performs decision-making. In addition, although it is closely associated with the decision-making element, the system must also have the knowledge about the dynamics of snooker. Snooker dynamics, especially the cueing dynamics, are also important from the system design perspective, as seen later in Chapter 5. The reviewed literature is broken down into four subsections: Game Strategy, Computer Vision, Manipulation and Snooker Dynamics.

Section 2.1 describes the intelligence that is needed to play the game, in the form of decision-making to select the best shot and the aspect of looking ahead into future shots. A brief coverage on the machine vision issues, with reference to the snooker system and other relevant systems, is given in Section 2.2. Section 2.3 reviews the possible manipulation solutions for a snooker robot by looking at similar manipulation methodologies - collectively known as nonprehensile, or graspless, manipulation - where objects are manipulated by methods such as tapping and pushing. A compilation on the physics of snooker is presented in Section 2.4.

2.1 Game Strategy

Players need to have both physical skill and strategy to succeed in a game. Games like chess are completely based on strategy. Snooker is not all about potting a single ball, but also finishing a frame successfully, pocketing all the balls in the given order. To accomplish this task, professional snooker players always look ahead of the current shot, i.e. they plan multiple shots in advance. They always have an idea as to which

ball to target in the next shot and so on. To play the next shot successfully it is important to leave the cue ball in an advantageous position on the table after the current shot. The configuration of the snooker table also makes it possible to play different types of shots (either straight, spin, kick or bank shots) and yet to have the same table outcome (e.g. in terms of the post-shot cue ball position). In other words, a given post-shot cue ball position can frequently be achieved in a multitude of ways. This makes the game more exciting and affords the player more flexibility in playing, but in turn makes the shot exploration procedure to select the best shot more difficult.

2.1.1 Selecting the Easiest Shot

Given a table state, this is all about selecting a shot (thus an object ball-pocket combination) purely on the basis of the ease of shot that maximises the chance of potting the selected object ball. Usually, amateur snooker players tend to think along these lines. In AI terms, this is known as *greedy local search*, because it targets the next best move without thinking ahead about which ball to pocket thereafter (not considering an overall optimum solution). A number of researchers have applied such greedy algorithms to formulate a playing strategy for billiards. Lin *et al.* **[2004]** have applied grey decision-making theory, which deals with the uncertainty and the knowledge incompleteness associated with a system, for their billiard robot. Lin *et al.* consider the distance between the cue ball and the object ball and that between the object ball and the pocket as well as the cutting angle required to pocket the object ball, for each object ball-pocket combination, d_{co}, d_{op} and $(180°\text{-}\gamma)$ respectively, as depicted in Figure 2.1. By considering a specific object ball and a given pocket, the analogy of Lin *et al.* introduces two ratios. A ratio of the lengths involved in the distance between the object ball and the pocket and the distance from the ball to its closest pocket. The second ratio involves the angles, and is defined as the fraction of the cut angle to the maximum possible cut angle of 90°.

When there are multiple object balls, Lin *et al.* **[2004]** introduce another length ratio: the one between the cue ball and a given object ball, and the shortest possible such distance amongst all the object balls. Finally, the three ratios are averaged to select an object ball-pocket combination to execute the shot, the shot that has the highest

average being the one selected. However, giving equal weights to all three factors, without evaluating their relative significance, in terms of the potting accuracy, is open to question. Relative weights for each of these factors need to be evaluated either through a theoretical analysis involving the shot geometry, or through a trial and error procedure by analysing the outcome of different shots played on the table. However, Lin *et al.* **[2004]** have not presented the results of the proposed idea in their paper, let alone compared it with other available methods.

2.1.1.1 Geometry-based Approach

Jebara *et al.* **[1997]**, in their work on aiding billiards players with wearable computers, resort to a strategy that assumes that the greatest difficulty presented to a player is in launching the cue ball in the ideal direction. The ideal launching direction is determined by assuming that the object ball will enter the pocket bisecting the entrance. It selects the best shot by calculating the maximum allowable angular deviation the player has (calculated by the shot geometry), which can still pot the ball, thereby selecting the shot which has maximum angular tolerance as the best one. This approach is based on the fact that the wearable computer is supposed to train amateur billiards players, who apparently have difficulties in directing the cue ball along the intended, ideal direction.

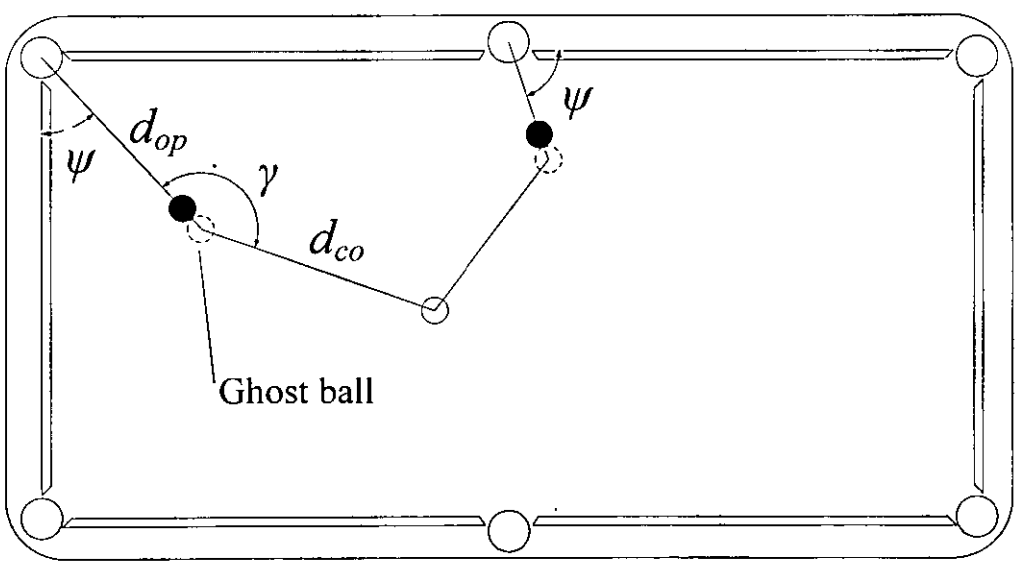


Fig. 2.1. Geometry of a direct shot

For the shot parameters as depicted in Figure 2.1, Shu **[1994]** defines the shot-difficulty index for potting a ball into any of the 4 corner pockets as, $\{k_1 d_{co}+k_2 d_{op}+k_3|\gamma\text{-}150°|+k_4|\psi\text{-}45°|\}$, k_1,.., k_4 being constants. For a centre pocket, the index is derived as $\{k_1 d_{co}+k_2 d_{op}+k_3|\gamma\text{-}150°|+k_4|\psi\text{-}90°|\}$. The justification provided in support of this formula arrangement is that professional players prefer γ closer to 150° than 180°, which is the angle for the in-line shot. The150° shot has the ability to allow the cue ball to move to almost any region on the table, as opposed to the in-line shot which positions the post-shot cue ball position only along the line of impact. For the in-line shot, to take the cue ball away from the line of collision requires excessive sidespin to be imparted to the cue ball by the cue.

2.1.1.2 Fuzzy Logic-based Approach

Chua *et al.* **[2002]** have used fuzzy logic to select the immediate best shot. d_{co}, d_{op}, γ (see Fig. 2.1) have been chosen as input parameters to the fuzzy decision-making element. Chua *et al.* also use three Gaussian membership functions for each of these inputs, which are denoted as easy, medium and hard in terms of difficulty. The fuzzy rule base consists of 27 rules, and the shot selection is based on the fuzzy outputs. Chua *et al.* **[2005]**, in their further work, decrease the number of rules to 24, by automatically omitting any ball-pocket combination where both the distance between the cue ball to the object ball and that between the object ball and the pocket are very large.

Chua *et al.* **[2005]** then use the zero-order Sugeno fuzzy model with the firing strength of the rules formed by the product operator. The *max aggregation* method is then used (i.e. *max-prod* operation) to obtain the output. Three singleton spikes are used as the output membership functions, which classify the shots as simple, intermediate and tough. Defuzzification is performed using the weighted average computation. Chua *et al.* then consider other types of shots with increasing difficulty e.g. the combination shot (involving two object balls), the kick shot and the bank shot. Whenever additional angles and distances are encountered in the complex shot configurations, they are modelled as fuzzy inputs, as described earlier. The algorithm searches within a type of shot, starting with direct shots, and if there are no shots to be

found, it starts searching in the shot type with the next level of difficulty, and continues the same procedure.

In their latest work, Chua *et al.* **[2007]** compare the results of the fuzzy approach with that of Jebara *et al.* **[1997]** and Grogono **[2005]** both of whom use a difficulty factor $\Delta = d_{co}.d_{op}/\cos^2\gamma$. Chua *et al.* present the comparisons for consistency between the two methods as applied to six different ball configurations on a billiard table. They also pass these table configurations to regular pool players (but their level of expertise has not been specified anywhere), to know which shot order the players would prefer in each of these six game states. Moreover, the results obtained by Chua *et al.* show that the decision-making has a close correspondence to the actual play, with 91.7% of the decisions being similar. The other algorithms with simple formulae, like those of Jebara and Grogono given above, also perform equally well. This seriously questions the necessity of this complex modelling. Alian *et al.* **[2004]** utilize the same fuzzy approach with only two parameters: d_{op} and γ. The justification, it is said, is because studies show that d_{co} does not have much influence on potting **[Alian *et al.* 2004]**. Nevertheless, no concrete evidence has been given in support of this argument.

2.1.1.3 Lookup Tables

In the work of creating an AI for computer billiards, Smith **[2006a]** uses a pool physics simulator named *Poolfiz* **[Leckie and Greenspan 2005]** to calculate the shot difficulty factors. Four shot parameters are employed; in addition to the 3 factors in the previous section, the object ball-pocket angle ψ is also considered (see Figure 2.1). This AI software for billiards, PickPocket, uses a lookup table to predetermine the probability of success for each feasible shot, in order to avoid costly run-time calculations. The shot difficulty parameters $\{d_{co}, d_{op}, \gamma, \psi\}$ are discretised, and sampling is used to fill in each table entry. For each set of $\{d_{co}, d_{op}, \gamma, \psi\}$ a probability of success is assigned after simulating a shot with these parameters in the *Poolfiz* simulator for 200 times repeatedly, noting the success rates.

2.1.2 Cue Ball Positioning

Experienced snooker players plan a few consecutive shots, and this is an essential element in winning any given frame. Hence, it is pertinent to consider the post-shot cue ball position (called position play), i.e. whether the cue ball will be in an advantageous position on the table to play the next shot and so on. If this aspect is not taken into consideration when executing the present shot, the player will have "snookered" himself for the next shot (i.e. the resulting table state will be such that the player cannot directly aim at any of the legal balls). If the shot is made without much deliberation, the cue ball may end up close to the cushions making the execution of the next shot difficult. In the worst-case scenario, if the post-collision cue ball motion is not carefully considered, after its collision with the object ball the cue ball can then fall into a pocket thereby awarding points to the opponent and losing the turn. The combined ball-table dynamics give a greater flexibility when it comes to position play. Figure 2.2 shows how the cue ball can be taken to diverse locations on the table by varying the power of the shot, and the amount and type of spin imparted onto the cue ball.

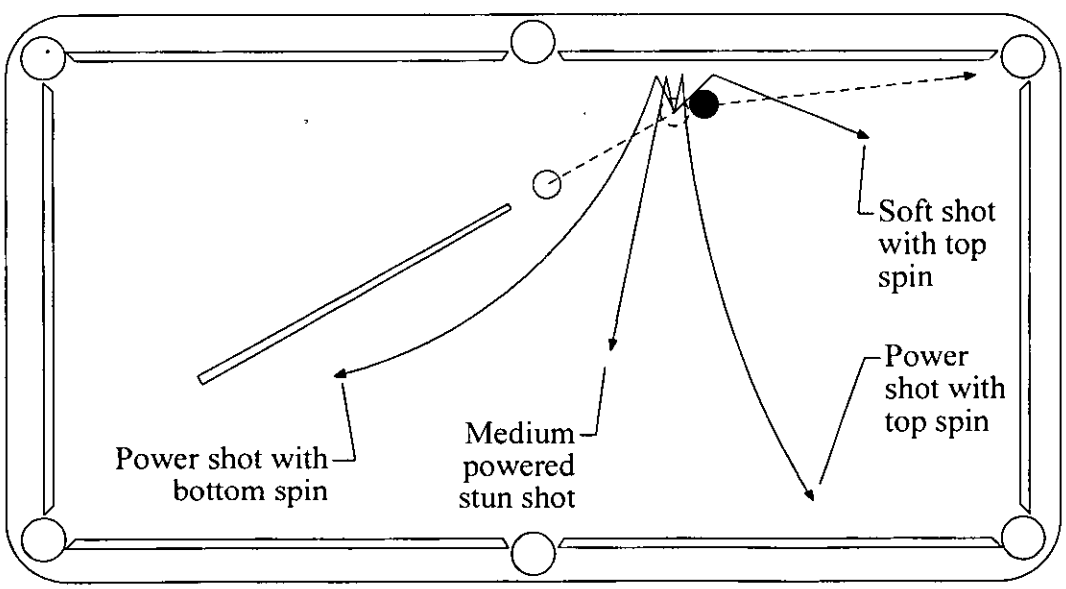


Fig. 2.2. Cue ball positioning with different types of shots

Dussault and Landry **[2006]** utilise optimisation techniques for positional play in their automated computer billiard player *PoolMaster*. Here they consider potting a single

ball by a straight shot and taking the cue ball to a given location. Using the sliding and rolling equations describing the motion of the ball after being hit by the cue, they calculate the initial linear and angular velocities (one linear velocity at a fixed direction and three angular velocities - four variables altogether) that are needed to propel the cue ball to a specified location. In order to calculate these four variables, an optimisation function is used, minimising the difference between the calculated final cue ball location and the desired location. Then they consider one ball collision to pot an object ball with a desired post-shot cue ball position as before. The constraint being that the cue ball should be present at two specific spatial points, one for impinging onto the object ball as required and the other being its desired final position when it comes to rest. Having this as the constraint, an optimization model is set up. Finally, the friction effects between the balls (i.e. collision-induced throw) are also added to the optimisation function. They use two optimisation software programs, Scilab and OPT++ (both use quasi-Newtonian iterative algorithms) to perform the computations. The optimisation procedure provides the linear velocity of the cue and the hitting point on the cue ball, so that the required amount of spin is imparted to it.

2.1.3 Evaluation Function and Look-Ahead/Search

In the previous section, the importance of position play and its significance in winning a frame of snooker is outlined. The next question is where to leave the cue ball after the current shot, since, as the table is a continuous domain, there are an infinite number of locations on the table where the cue ball can be positioned. Thus, for any given location on the table, there needs to be an evaluation of the merits of that position. Once this evaluation is formulated for the whole table surface, the problem of finding the best spot to reposition the cue ball can be solved by searching the table space in order to maximise the evaluation formula. This will result in the formulation of an overall strategy to play the game.

2.1.3.1 Evaluation Function

"An evaluation function returns an estimate of the expected utility of the game, for any given position in the game" **[Russell and Norvig 2005]**. In snooker, for a given

state (i.e. positions) of the balls on the table, the evaluation function must be able to tell how useful that state is to the player from the perspective of winning the frame. Evaluation functions work by calculating the features/patterns of the state. In chess, the possible features can be, for example, the number of pawns possessed by each side, and king safety. For example, the Deep Blue evaluation function uses 8,000 different patterns **[Campbell *et al.* 2002]**. Then each of these features is assigned a numerical value (using some kind of *a priori* knowledge of the domain) and a weighted linear function is assigned, adding up all the features.

In chess, the game states are discrete and the actions are completely deterministic. However, in snooker, there are an infinite number of positions on the table where the balls can be positioned, i.e. it has a continuous nature. Also, for a cue ball-object ball combination, there are an infinite number of distinct shots to be considered (for example changing the cue ball direction by a small amount, imparting different types of spin). Hence the snooker domain, unlike chess, is continuous. In addition, due to inaccuracies in the sensing elements, such as the camera, 'noise' is introduced into the estimation of the state of the balls on the table, which, in turn, introduces a small degree of randomness into the artificial system designed to play snooker. The current evaluation functions for billiards/snooker are based on the number of quality shots available from a given cue ball position, thus giving greater flexibility to a player. Smith utilises the sum of the difficulties of the total number of legal shots available from a given cue ball position **[Smith 2006a]** as the evaluation function for that position. Dussault and Landry **[2007]** use both the total sum and the average value of the shot difficulty for the analysis performed in their artificial intelligence software, PoolMaster.

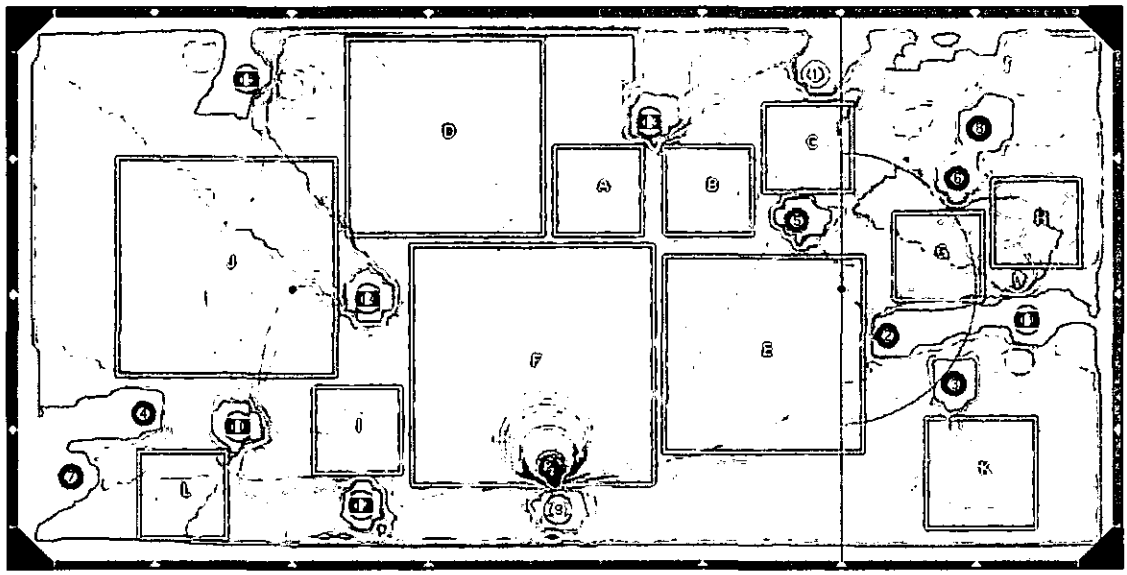


Fig. 2.3. Outline of the best regions for the cue ball positioning using the sum measure for a given ball configuration in pool **[Dussault and Landry 2007]**

Dussault and Landry **[2007]**, in addition, create a global map of difficulty for a given configuration of the balls on the table, using contours, which specify the regions where the cue ball would be favourable to play the successive shots (see Figure 2.3). In their further work, Landry and Dussault **[2007]** also use a pre-computed table where the table area is discretised as grids, and the cue ball and the object balls are placed in each of these grids in turn and a shot is taken, all within the simulated environment of PoolMaster. Furthermore, for each combination of the cue ball and object ball positions, the shot parameters, in the form of cue velocity and the striking point on the cue ball, are also changed. All of the results are then stored in tabular format. For a given situation, to find the optimal shot parameters, before the optimisation routine is executed, the starting values for the routine are found from the closest matching values in the look-up table. Specifying a closer starting point for the routine drastically reduces the time taken for the optimisation.

Smith **[2006b]** also proposes to include the *centredness* of the balls (a ball in the middle of the table has the advantage of being easily pocketed in all the pockets) in the evaluation function. Moreover, according to Smith **[2006b]**, the ball *clusteredness* (which is an impediment to successive potting of the balls) can also be included in the evaluation function to give it a more realistic value **[Smith 2006b]**. In addition to these factors, other aspects such as the close proximity of the balls to the cushions

also have an influence on potting. Incorporating them in a suitable way may lead to a better strategy.

2.1.3.2 Look-Ahead/Search

Look-ahead is very important for any game that is turn-based, and look-ahead is accomplished by searching for possible future game states that can lead to a win. In pool, professional calibre 8-Ball players try to visualize the entire run of eight balls at the start **[Leckie and Greenspan 2007]**.

Chess and checkers are considered as "standard" games in the traditional analysis, and very successful computer chess programs have been created, culminating in Deep Blue's win against the world chess champion, Gary Kasparov. The strength of these programs depends on their power to search for the best moves, which has apparently been aided by today's high performance computers. For chess and checkers programs, the *mini-max* game trees are used to analyse different possible future game situations; the search algorithm employed is *alpha-beta* **[Russell and Norvig 2005]**.

Contrary to chess and checkers, snooker is a game that has a continuous nature, i.e. there are an infinite number of shots to consider for any given state of the table. This makes the search difficult and necessitates the use of different techniques to search for the best shot. Snooker is a non-deterministic game where, even if one plays two identical shots, the table outcomes can still be different because the physical parameters of the system are subject to change with time, however small they are. Especially, when an artificial system plays snooker, the errors involved in the sensing/ computer vision elements and the actuator will always introduce deviations from the ideal expected outcome. For this reason, and to emulate a real-world scenario, some computer billiards programs (these are completely deterministic), are superimposed with Gaussian noise on their parameters to give a more realistic experience to the users **[Greenspan 2005]**. Hence, the snooker environment has a stochastic nature. This makes it difficult to do the search in a purely analytical way. However, the strong point for developing a good search algorithm for snooker is that, as opposed to games like chess, which alternate the turn to the opponent after every move, it is possible in snooker to clear the table straight away without giving a chance to the opponent, thereby winning the frame.

The *Expectimax* algorithm is used for the search of stochastic domains, as in the game of backgammon where the dice-rolling introduces uncertainty. Under such environments of a stochastic nature, the **-Minimax* algorithm can be used to prune the search tree, reducing the run-time. Even then, backgammon, having a finite number of nodes, has discrete states, compared to snooker, that present a continuous, stochastic domain. Hence, for any given parent node in the search tree, in snooker, there will be an infinite number of children with minuscule probabilities. Smith applies probabilistic search to solve this problem **[Smith 2007]**. Every shot is classified as a success or a failure; a physics simulator is used to get a value for the probability of success. Only successful shots are expanded in the game tree. This method is a very crude abstraction of the game situation, because, even though not all the successes are equal, they are given equal value by the algorithm.

Monte Carlo methods have also been applied to get a random sampling in the range of possible outcomes of a shot, and they use the **-Expectimax* method to search among these possibilities **[Leckie and Greenspan 2007]**. The higher the number of samples, the better the performance.

2.2 Computer Vision and Ball Tracking

Computer vision is needed to sense the ball locations on the table and so will be the primary sensing element of a robotic snooker system. Snooker requires a 2-D visualisation process in order to extract the ball centres since the ball centres are always constrained to a plane parallel to the table surface at a fixed height, equal to the ball radius. Only the jump shots violate this condition, but they are very rare in the game and are outside the scope of this work. The existing systems to play snooker/pool use overhead-mounted cameras looking vertically downwards to visualise the ball positions on the table. In addition to an overhead camera, the Bristol University system also consisted of an eye-in-hand visual servoing unit to compensate for the inaccuracies associated with the overhead camera **[Shu 1994]**. The Deep Green system **[Greenspan 2006]** also uses a second camera whose image plane is

perpendicular to the longitudinal axis of the cue. Jebara *et al.* **[1997]** use a head-mounted video camera as the vision element of their wearable computer that trains amateurs. All the above-mentioned configurations image the table area in order to locate the static ball locations, so that a decision can be made on which shot to play next, as elaborated in Section 2.1. In the robotic systems, the cue ball location together and the shot parameters are supplied to the robot controller for the robot to take aim and execute the shot. To establish the performance of the robot, the post shot static ball positions on the table are also observed. However, the literature review shows that there were no serious efforts to track the balls continuously. Continuous ball tracking can provide some useful inputs to the system, such as the velocity variation of a ball, possible deflections due to the spin shots, cue squirt or any other variations due to the table surface, like the nap of the table cloth **[Williams 2002]**. This author believes that the ball tracking can better inform of robot controller about the actual dynamics of the ball motion on the table. The tracking can either be performed whilst the robot is playing a shot, i.e. in real-time or, whilst 'off-line', it may be by tracking human shots. In a similar project, on developing an air-hockey-playing robot, Bishop and Spong **[1999a, b]** tracked the puck continuously on the air hockey table, so that their robot could effectively strike the puck.

The primary reason for advocating ball tracking is the lack of theories to describe the dynamics of snooker, especially the collisions in the presence of friction, as described later in this chapter. Moreover, when new theories are put forward for an aspect of snooker dynamics the validation can only come through an experimental procedure and this necessitates some form of ball tracking. In the opinion of this author, a spin-tracking element will also be an essential part of the experimentation.

To determine the fundamental physical parameters, such as the coefficient of restitution, the Bristol University project used talcum powder in certain table areas to track the balls, but this is not very efficient and prone to errors **[Shu 1994]**. There are also other variations in the game, such as the drifting effect due to the natural line of the nap of the table cloth **[Williams 2002]**, ball-pocket interactions etc. A visual tracking system enables the system to learn and adapt to any unusual behaviour found in the snooker dynamics.

2.2.1 Lighting

As mentioned earlier, both global and local (eye-in-the-hand) cameras were used in the snooker/billiards robots. However, the global vision of the table presents a number of technical problems. First, it necessitates a highly uniform and diffused lighting, because any focussed lights would affect the image of the balls, which are highly reflective. The Bristol system used a monochrome camera for the global imaging, with tungsten halogen dichroic reflectors **[Shu 1994]**. The dichroic reflectors, which are used behind a light source, allow the visible light to go forward and pass out the radiated light to the rear, this helps avoid heating up of the object being viewed. Reflectors were kept in an 8x5 matrix above the snooker table, after considering the luminance map of a single source, the distances between the adjacent reflectors are chosen so as to have a null luminance trough between them. The robot on-board camera in the Bristol system has its own miniature lighting with an identical light source to the one above. The table lighting arrangements of the other robots are scarcely described in the literature.

2.2.2 Camera

To image the equal sized, coloured balls found in snooker, it is straightforward to go for a colour camera. However, a monochrome camera was used by Shu **[1994]**. Initially, a grey scale map for a ball with given colour was plotted. Using this map, a histogram of the average grey-scale of all the balls using 100 repeated trials was obtained and this histogram was used to distinguish the colours of the various balls. A classifier based on the average grey-scale value of a colour identifies a particular ball. A colour camera would dispense with most of this trouble. The wearable computer system used a colour camera with an RGB based probabilistic colour model, with data clustering, to distinguish different coloured pool balls (pool has both solid-coloured and striped balls) **[Jebara *et al.* 1997]**.

2.2.3 Image Processing and Feature Extraction

Before a camera can be used to make any measurements, it must be calibrated in order to relate the image parameters to the real world parameters. For the stationary overhead cameras frequently used in the robots, there is no relative motion between the camera and the table, which usually represents the real-world coordinates. Camera calibration involves determining both the pose of the camera with respect to the scene (called the external/extrinsic parameters), and the internal camera parameters (also known as the intrinsic parameters), such as its focal length and the image magnification in the sensor. Generally, the lens distortion is modelled by a polynomial that should be added to the above-mentioned model to make the calibration complete **[Gennery 2006]**. There are pattern based calibration methods, where a calibration object with a known geometry is moved while the camera is kept stationary. A camera can also be calibrated by what is known as self-calibration, where the camera is moved in a static scene, and the rigidity of the scene provides the constraints that are necessary to estimate the intrinsic and extrinsic parameters of the camera **[Zhang 2000]**.

Once the image is corrected for its distortion, it can be used for ball identification (i.e. identifying the colour of the balls) and then locating the centres of the balls. Chua *et al.* **[2003]** used an image mask to isolate the playing area of the table with the aim of increasing the speed of image processing. From the colour camera input (the RGB image), Chua *et al.* used only the R and G components for image processing (they state that these two components are sufficient to provide the required results). They then apply an intensity adjustment on these two components separately, to increase the contrast, before applying the Sobel operator to detect the edges which correspond to object boundaries. The two processed R and G components are combined to get a binary edge detection image. Then, Chua *et al.* **[2003]** use the Circular Hough Transform to locate the ball centres.

Long *et al.* **[2004]** have corrected for the radial distortion of the lens, before applying a threshold to produce a binary image. They then use a connected component algorithm to eliminate the noise, and to eliminate significantly small blobs to locate the balls. It should be noted that there are other robust filters, e.g. the median filter, to

cope with the noise present, even though these are computationally expensive. The ball centre is then located by a least square estimation, with the radius of the circle known *a priori* **[Long *et al.* 2004]**.

Other researchers have used a probabilistic colour model to visualize and differentiate the table area within the general field of view **[Jebara *et al.* 1997]**. They initially obtain a number of sample pixels of the table cloth under different lighting conditions (their work concerns a moving imaging system, i.e. a wearable computer) and use data clustering to get a plot of the RGB distribution and use Expectation Maximization (EM) to find a probability distribution for the colours of the pool table. As a new image is acquired, Jebara *et al.* evaluate each pixel for its likelihood of being a part of the table, and if it is above a certain threshold value, it is labelled as the table. The edges of the table are also found using the EM algorithm. Then a probabilistic colour model is trained to recognise the pockets. Since the work is concerned with an 8-ball Pool game, a probabilistic colour model is also used to detect the balls that are both solid and striped. Denman *et al.* **[2003]**in their work concerned with extracting useful information from the video broadcasts of snooker, have used colour-based segmentation to locate the table area whilst using suitable masking for players; the Hough Transform is used to detect the edges.

There are also studies that deal with self-shadows and occlusions of balls under natural lighting conditions. For example, D'Orazio *et al.* **[2004]** have employed a special operator for semicircle detection in conjunction with a neural classifier to identify a football within a football field.

2.2.4 Object Tracking with High Speed Imaging

Earlier, it was argued that the ball tracking is necessary to play the game effectively. Of late, object tracking has been widely used in ball sports. Hawk-Eye is one good example where a six-camera system is used to track the balls in cricket, tennis, baseball and snooker for television broadcasts **[Pingali *et al.* 2000]**. Each of Hawk-Eye's six synchronised monochrome cameras operates at a speed of 100 frames per second (fps), which uses acquired data from tracking points to predict subsequent ball

trajectory and bounce whenever the ball motion is blocked, as in the case of leg-before-wicket decisions in cricket **[Davis 2009]**. The accuracy of Hawk-Eye is reported to be between 2 to 5 mm. There are also other vision- and radar-based tracking systems that measure the magnitude of ball spin, using the features on its surface, like the seams **[Griffiths *et al.* 2005]**.

2.3 Nonprehensile Robotic Manipulation of Objects

In Chapter 1, while providing an outline of the current systems to play snooker and pool, their configurations have been summarized. A review of relevant manipulation methodologies that will be useful for a snooker robot is given in this section. The manipulation schemes that are collectively known as nonprehensile manipulation methods are considered.

Nonprehensile Manipulation

Nonprehensile manipulation is defined as the manipulation of an object without grasping it (hence also known as graspless manipulation) **[Mason 1999]**. Figure 2.16 depicts a typical situation encountered in graspless manipulation where an object is manipulated by pushing for a sorting operation by two robots. A clear advantage of such a scheme is that it does not require very large robots to cover the whole workspace of the object. In snooker, the very nature of the game only allows for nonprehensile manipulation, an impulse-based manipulation to be specific, of the cue ball by the cue. Hence, nonprehensile manipulation methodologies will be extremely useful for a snooker robot. Nonprehensile manipulation was developed as a substitute for the pick-and-place manipulation, which has limitations because of the sizes of the objects that can be manipulated and the size of the workspace.

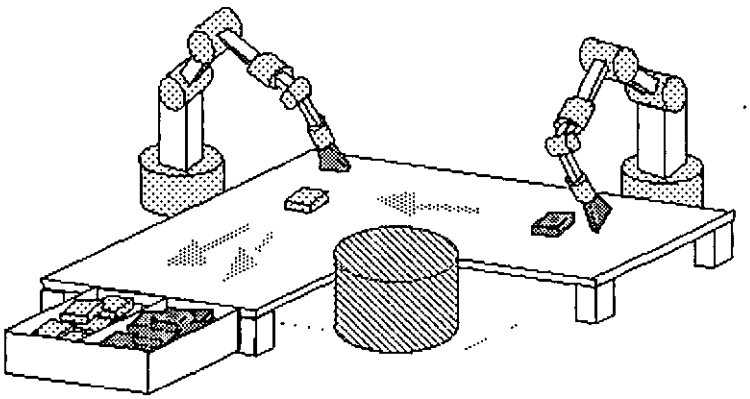


Fig. 2.4 An example of nonprehensile manipulation **[Zhu *et al.* 2006]**

Without grasping the object, a nonprehensile manipulator with fewer degrees of freedom (DOF), uses the gravitational, centrifugal, Coriolis and frictional forces acting on the object as virtual motors to control more DOFs of the object. The object's extra motions are exhibited as rolling, slipping and free flight **[Lynch and Mason 1999]**. For example, the cue-driving action, which is equivalent to a single DOF pusher, by changing the cue velocity, can theoretically position the cue ball on the table, a space with two DOF, which is one more than that of the manipulator. Manipulating in a higher dimensional space is essentially achieved by allowing the object to move relative to the robot. Many other manipulation methods such as throwing, batting, catching, orienting/reorienting parts, and manipulation by impulses, also come under this category.

Nonprehensile manipulation presents other problems of its own, when compared to grasping manipulation. Grasping manipulation, as in a pick-and-place robot, requires a description of a path for the end effector to follow. This is accomplished by means of programming the individual joints accordingly, and their control is a classical closed-loop problem. However in nonprehensile manipulation, the object is released from the manipulator at a specific time and the object's control thereafter is not with the manipulator. After the object loses contact with the manipulator, its control is taken over by the virtual motors, i.e. the gravitational, frictional or its own inertial effects. From the perspective of the manipulator, the whole scheme is analogous to that of an open loop system, with no feedback control. This 'open loop' nature of nonprehensile manipulation requires a detailed preplanning that considers every aspect of the dynamics of the object motion.

Although there are numerous nonprehensile manipulation methods, as outlined above, only the ones where a force/impact is applied to an object and it is then being allowed to slide on a planar horizontal surface are considered here. Here the object, in its 'free' motion, is only under the influence of frictional and inertial forces. This type of problem has been analysed by many researchers. Impulsive manipulation **[Huang *et al.* 1995, Partridge and Spong 2000]**, releasing manipulation **[Zhu *et al.* 2006]**, tapping manipulation **[Huang and Mason 2000, Han and Park 2001]**, multi-agent dynamic cooperative manipulation **[Li and Payandeh 2003 a, b]**, pushing manipulation **[Rezzoug and Gorce 1999]** are some examples. In all these manipulation operations, two distinct situations may be encountered, one where an algebraic analysis of the object dynamics is completely possible, and others where this analysis is only partially possible or completely impossible.

2.3.1 Planning the Object Motion

As said earlier, nonprehensile manipulation requires extensive pre-planning. The type of manipulation under consideration consists of two dynamic phases, the robot's actual interaction with the object, and the 'free' motion of the object. Hence, the planning also consists of two phases: the inverse solutions for the 'free' motion of the object, and the inverse dynamics for the manipulator-object interaction. The initial requirement is to get the inverse solution, i.e. determining the initial velocities, both linear and angular, of the object for a given requirement of object transfer, such as its final position and pose, as well as the desired path of the object. Secondly, the manipulator interaction parameters must be determined, i.e. the amount of impact and the suitable contact point on the object that is necessary to generate the required initial velocities of the object that will eventually lead to the desired final object configuration. Certain final object pose configurations may not be possible due the current pose of the object, or because of the limitations of the robot; such cases have to be clearly distinguished during the planning phase. The limitations are also due to the geometry of the object being manipulated and the environment within which the whole operation takes place. As a simple example, in a single DOF impact and planar sliding problem, the maximum reachable distance is determined by the maximum

power of the manipulator, the friction and geometry (e.g. the horizontal tilt) of the surface, and the object geometry.

The inverse solution for the 'free' motion problem is tedious whenever the object motion problem is not fully, or maybe even partially, algebraically describable, otherwise, the inverse solution is very straightforward. For the former, one has to resort to some other methods. Huang *et al.* **[1995]** followed a numerical approach, and used the properties of displacement monotonicity with respect to the initial velocities, to obtain the inverse solution for a planar disk sliding on a table. To orient a triangular planar part on a surface, Han and Park **[2001]** made use of a numerical algorithm to obtain the inverse solution. The numerical algorithm is based on the qualitative motion characteristics, such as the monotonicity between parameters. However, Mason **[1999]** utilized optimisation to find the trajectories of a polygonal part after it has been struck, thereby establishing an inverse solution. In their work on dynamic cooperative manipulation, Li and Payandeh **[2003a]** used a quasi-Newtonian method as their optimisation routine. Their further work adopts game theory and neural networks to overcome the uncertainties present in the environment **[Li and Payandeh 2003b]**. Zhu *et al.* **[2006]** utilized two iterative learning control schemes to solve the problem of inverse sliding for the planar objects. Matsushima *et al.* **[2005]** made use of inverse maps (input-output maps) and locally-weighted learning for the inverse problem encountered in their table tennis robot, to determine the paddle's velocity and its angle of hit. By observing the ball hit by the opponent as it passes a predetermined vertical plane on the opponent's side, this mapping also predicts where it will hit the table on the robot's side and predicts its velocity as well.

The impact dynamics between the manipulator and the object are always a complex phenomenon that is challenging to model and control **[Li and Payandeh 2003a]**. In this respect, the Routh impact model and the Newtonian model for the impacts are generally used. Routh's model is based on a graphical analogy of the impact configuration and is considered more accurate **[Wang and Mason 1992]**.

2.3.2 Manipulation Limitations

Not all the initial velocity combinations (both linear and angular) are always possible for a given object. These depend on the geometry of the manipulated object and the manipulator, material properties such as the friction between the manipulator and the object, support conditions of the object, etc. For example, in snooker, for miscueing not to occur the cue must impinge the cue ball close to the stun point (see Section 2.4.1), if the cue ball is struck at its periphery it will barely move, and definitely not along the expected direction. Spong **[2001]** has analysed for the reachability of an air hockey puck struck by a puck. The reachable velocity of the puck and its impact controllability are derived in terms of the friction coefficient and the coefficient of restitution between mallet and puck using the Routh impact model. The main difference between an ice-hockey puck and a snooker ball is that the puck can only have spin about the vertical axis, whereas a ball generally has the potential to spin in three dimensions.

2.4 Snooker Dynamics

Snooker presents a classical physics problem, and understanding it is vital for any artificial system to play the game. Skilled snooker players may not understand the dynamics in terms of physics, but having spent enough time with the game, they know the intricacies of the game, by way of experience. A robotic system can be designed to take this approach and 'learn' the game, using a machine-learning paradigm, such as supervised learning with neural networks, where the controller learns from the examples provided to it **[Jang *et al.* 1997]**. However, if a system-dynamics-based solution is pursued to determine the ball trajectories, the physics involved in the game has to be clearly understood.

The first published analysis of billiards physics was from Coriolis in 1835 **[Nadler 2005]**. Since then it has been a topic for teaching and research in physics. This section looks at different phenomena that are involved in snooker dynamics. They are: the

interaction between the cue and the cue ball (i.e. cueing), the rolling of the ball on the table, impact between two balls, the ball's collision with a cushion and the dynamics of ball spin.

2.4.1 Cueing

Cueing, which is the manoeuvring of the cue and then hitting onto the cue ball, is the sole interaction of the player with the system dynamics, and hence it is very critical from the player's point of view. This is a very subtle operation and involves a considerable amount of skill. Here, the long, slender cue imparts an impulse force onto the cue ball forcing it into motion; the impact is delivered, theoretically, through a point contact. In order to facilitate such a contact the cue has a rounded tip.

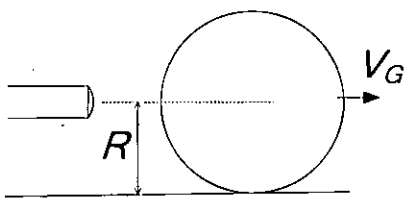


Fig. 2.5. A stun shot - no angular velocity imparted by the cue

As shown in Figure 2.5, when the hit is horizontal and through the centre of the cue ball (this is called a stun shot), the initial velocity, V_G of the ball can be determined from the impulse, I, by

$$I = MV_G$$

where M is the mass of the cue ball and I is the integral of the force between the cue and the ball over the time of impact. $I = \int F dt$

The amount of impulse is always difficult to calculate, because the variation of the impact force with time is not known. A simple solution is to measure the force using a transducer with the respective time stampings and then to numerically integrate the force over time to obtain the total impulse. If an average force F_m (this can be the time-average reading of the force transducer) can be defined during the time of impact, Δt, it can be shown that, $V_G = \frac{F_m . \Delta t}{M}$. However, cueing is not always as

simple as this and whenever the cue is not maintained horizontal or if the line of strike does not go through the ball centre, the ball starts to spin about its centroid (this spin is also known as 'English'). Depending on the position of hit on the cue ball, spin is called top, bottom, left or right (Figure 2.6). It is also possible to get a composition of two types of spin as shown below in Figure 2.6, by selecting the striking location appropriately.

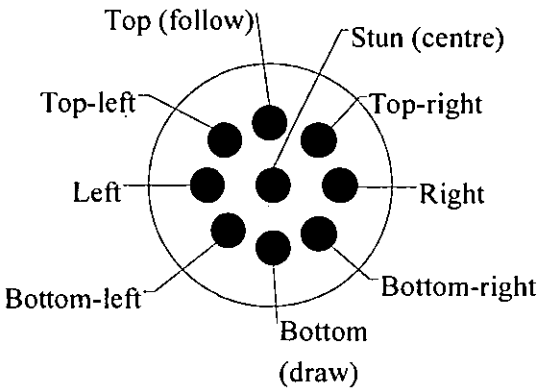


Fig. 2.6. Types of spin with the respective striking areas on the cue ball

2.4.1.1 A Generalised Model for Cueing

Next, a more generalised situation of cueing is considered, where no constraint is imposed as to how the cue is being held while striking the ball. An analysis can be easily performed for this configuration without considering frictional effects due to the table surface and between the ball and the table **[Salazar and Sanchez-Lavega 1990]**. However, de la Torre Juarez in his work on cueing considered the effect of friction from the table during impact **[de la Torre Juarez 1994]**. The presence of friction causes a friction impulse, and makes the analysis complex. The following is an abridged version of the detailed analysis of de la Torre Juarez **[1994]**, though it is a repetition of the published material, it is given here it as it gives a completeness to this review on billiards literature (only the essential equations, and derivations are given here).

Assuming that the time of impulse is very small, and the distance the ball travelled during the impulse is small and that the direction of ball velocity remains constant during the impulse.

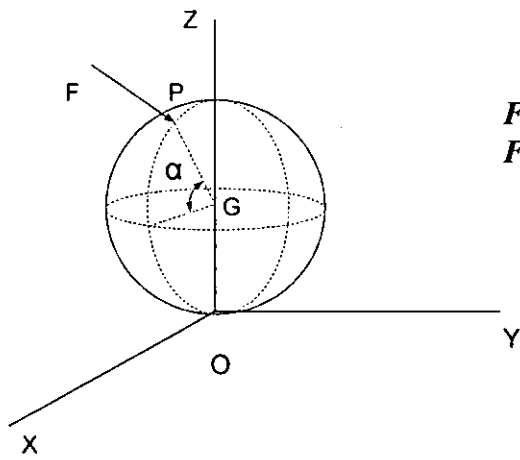


Fig. 2.7. A general cueing situation

Using the linear impulse-momentum equation, and also the equivalent equations for the angular motion of the ball, and referring to Figure 2.7,

$$MV_x = I_x + I_{f,x}$$

$$MV_y = I_y + I_{f,y}$$

$$MV_z = I_z + F_N$$

$$2/5MR^2\,\omega_x = R(-I_y \mathrm{Sin}\alpha + I_{f,y})$$

$$2/5MR^2\,\omega_y = R(I_x \mathrm{sin}\alpha - I_z \mathrm{cos}\alpha - I_{f,x})$$

$$2/5MR^2\omega_z = RI_y \mathrm{cos}\alpha$$

Where, N is the normal impulse between the table and ball at contact point O, and I_f stands for the frictional impulse along the table. ω is the angular velocity of the ball about G. The linear velocity V corresponds to the centre of gravity G of the ball ($V_G = \{V_x, V_y, V_z\}$). ω_x, ω_y and ω_z are the angular velocities about the X, Y and Z axes respectively. Unless the ball jumps up from the table (this is what happens in a jump shot), $V_z = 0$ and $F_N = -I_z$. One notable observation from these equations is that even if the cue points towards the centre of the ball (as considered initially), due to the effects of friction from the table, there will still be some angular momentum imparted to the ball.

Case I: Rolling motion

When the ball rolls, at O, $V_O = 0$, and it can be derived that $V_x = R\omega_y$, $V_y = -R\omega_x$ and using the above equations with the condition involving μ_s , the sliding coefficient of friction between the ball and the table, $I_{f,x}^2 + I_{f,y}^2 < \mu_s^2 F_N^2$.

Solving the above equations, if the ball is to roll, $\frac{\sqrt{(49\mu^2 + 25\cos^2\alpha)}}{(5\sin\alpha - 2)} > |\frac{I_y}{I_z}|$.

Case II: Slipping motion

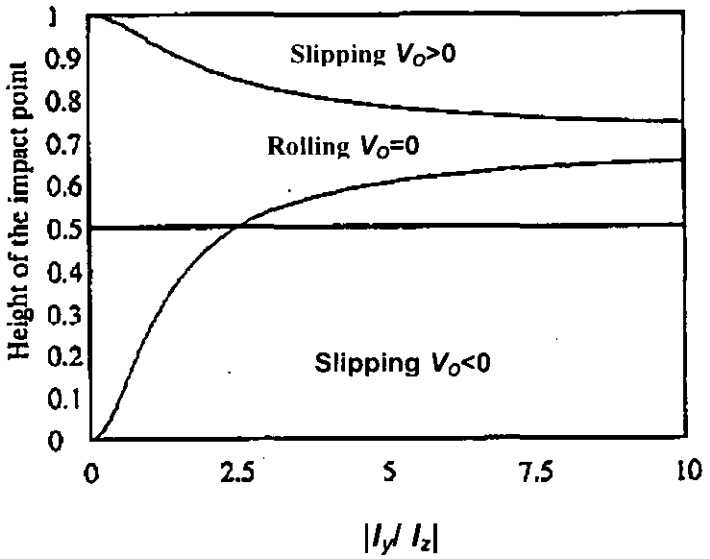
Slipping is more complicated than rolling as the condition $V_O = 0$ is no longer valid. But now, the friction impulse is given by, $\vec{I}_f = -\mu_s F_N (\frac{\overrightarrow{V_O}}{V_O})$.

The impulse-momentum equation can be expressed written in vector format as, $M\vec{V}_O + M\vec{\omega} \Lambda \overrightarrow{OG} = \vec{I} + \vec{I}_f$. Using this and the velocity relationships between points O and G, the velocity of O (i.e. the sliding velocity) is derived as, $V_O = (1 + \tan^2 \Phi)^{1/2} (I_x / M - R\omega_y) - \mu_s |I_z| / M$,

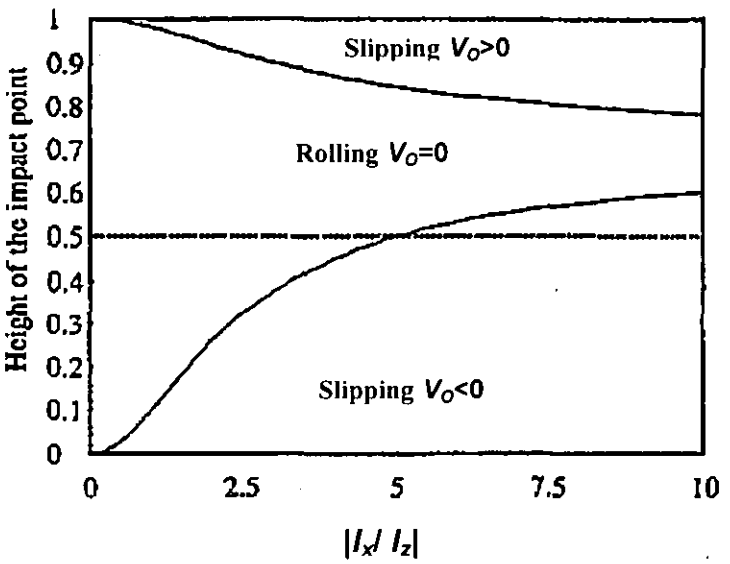
where, $\omega_y = \frac{5}{2mR}(I_x \sin\alpha - I_z \cos\alpha - I_{f,x})$ and Φ is the angle V_O makes with the OXZ plane and is given by, $\tan\Phi = \frac{(2 - 5\sin\alpha)I_y}{(2 - 5\sin\alpha)I_x + 5I_z \cos\alpha}$. V_O is useful in deciding whether the ball slips or rolls after being struck. Figures 2.8 (a) and (b) illustrate the conditions that give rise to different types of ball motion for different values of {I_x, I_y, I_z}, where instead of angle α, the height of the hitting point on the ball, which is related to R by the formula $R(1+\sin\alpha)$, is used.

But the motion of the ball is given described by the velocity of its centre of gravity, V_G, and can be calculated from, $\overrightarrow{V_G} = \overrightarrow{V_O} + \vec{\omega} \Lambda \overrightarrow{OG}$. The direction of V_G with the OXZ plane, θ, is given by,

$$\tan\theta = \frac{6\mu_s \sin\Phi + (1 - 5\sin\alpha) - (I_y / I_z)}{6\mu_s \cos\Phi - (1 - 5\sin\alpha)(I_x / I_z) + 5\cos\alpha}$$



(a)



(b)

Fig. 2.8. Influence of height and impulse components on the subsequent motion of the cue ball **[de la Torre Juarez 1994]**

Slipping is the phenomenon that makes curved shots possible. Conversely, when a ball rolls on the table it can only move along a straight-line.

There is a limit for the reachable velocities for the cue ball, both linear and angular, as there are limitations on the cue speed (whether it is manipulated by a human or a robotic system), the amount of cue-cue ball friction and the geometrical constraints by way of the shape of the cue and the cue ball. Nevertheless, no such study substantiated by an experimental procedure exists in this regard. To draw a parallel to a future study, the work of Spong **[2001]** on the air-hockey puck striking can be used. For example, Spong examines the reachable velocities for the air hockey puck, which is relatively simple in that it only has one rotation and two linear velocity components. However, in snooker, there are three linear and another three rotational velocity components making the analysis and experimentation very complicated.

2.4.1.2 The Massé Shot

Even for the shots played with English (sidespin), the trajectory of the ball is still rectilinear. But when the cue is highly elevated from the horizontal and kept close to the vertical and struck downwards on the ball powerfully, the cue ball starts to move along a curved path (see Figure 2.9), and the shot is called a massé. Massé occurs due to the 'sideways spin' that makes the ball rotate about its initial frontal velocity axis as depicted in Figure 2.9 (or the 'Aiming line' as shown in Figure 2.10).

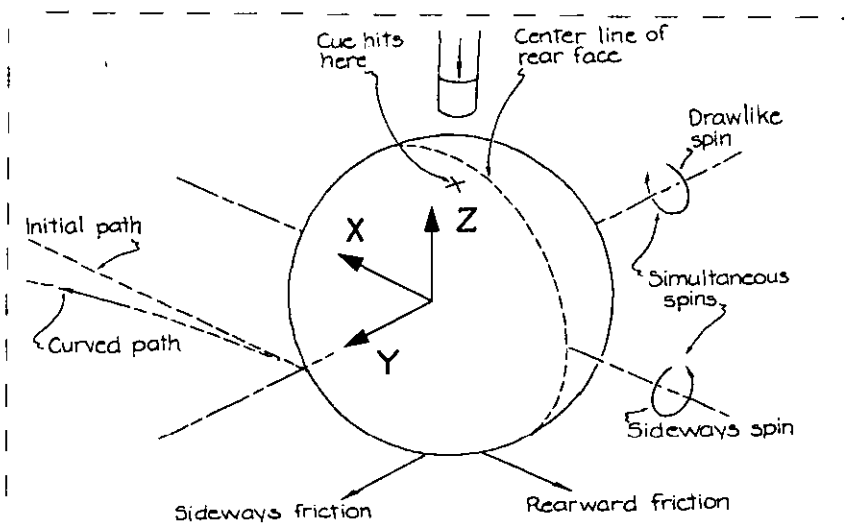


Fig. 2.9. Factors involved in a massé shot **[Walker 1983]**

This shot is very useful when the cue ball is in an awkward position and no legal straight shots are possible. When played to a collision with a cushion, the massé shot

generates quite astonishing motions on the cue ball (for example see Figure 2.10). There have been several attempts to outline the physics of the massé shot e.g. Walker **[1983]**. However, mathematical models describing massé have not been found in the billiards literature.

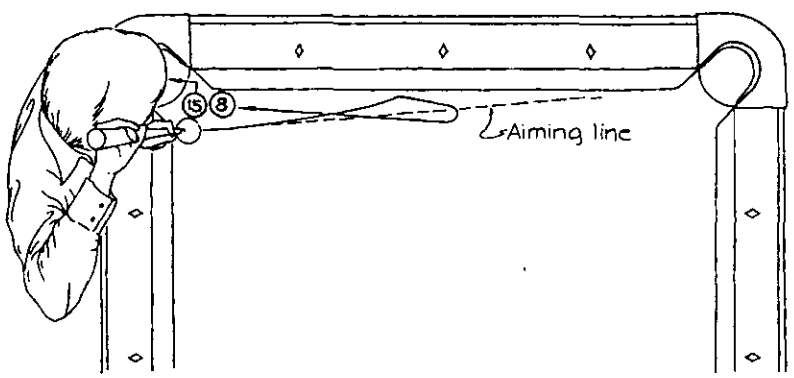


Fig. 2.10. A complex massé shot sinking both 15-ball and 8-ball in pool
[Walker 1983]

The massé shot supplies spin about two horizontal axes, X and Y, and has a negligible amount of sidespin (spin about the Z-axis) (see Figure 2.9), axes definitions are similar to that in Figure 2.7. With the cue almost vertical, the player strikes downward on the side of the cue ball. The horizontal part of the stroke determines the initial path of the ball. The sideways spin (about the X-axis) gives rise to sideways friction force as shown in the Figure 2.9. This friction force, which is perpendicular to the ball motion, makes the ball trajectory curve. The ball moves along a parabolic path until it starts to roll. Once the ball has stopped slipping, and starts to roll, the trajectory becomes linear again (see the final part of the ball trajectory in Figure 2.10). The trajectory of a cue ball, under the massé conditions, can be derived mathematically. Hopkins and Patterson **[1977]** have done similar work on the paths of a bowling ball, where they derive a mathematical description of the curved trajectory.

2.4.1.3 The Cue Ball Deflection ("Squirt")

Whenever the cue is struck toward one side of the cue ball, in order to impart sidespin, the cue ball, in general, starts to move in a direction that is slightly different to the line of approach of the cue. This phenomenon is called cue squirt. Inadequate knowledge of cue squirt is identified to be the major problem with amateurs **[Jewett 1994]**. Cue squirt happens because the ball attempts to move along the direction of the common normal of both bodies through the impact point. However, the action of friction between the cue and the cue ball usually reduces cue squirt (see Figure 2.11). Shepard **[1997]** has analysed the physics of cue squirt elaborately. This theory says that apart from the factors shown in Figure 2.11, i.e. the offset distance b, the equivalent inertia of the cue-cue tip combination (called the 'endmass') also affects the angle of squirt **[Shepard 1997]**. Shepard's theory also predicts that the squirt angle α, increases with the value of the 'endmass'. The implication of the 'endmass' effect means that the grip of the player affects cue squirt, as the 'endmass' would be high if a tight wrist is maintained during cueing, and will be low for a loose wrist condition. The main drawback of Shepard's theory is that the friction values between the cue and the ball have not been used, thus the forces F and N (see Figure 2.11) are kept as independent parameters throughout the analysis.

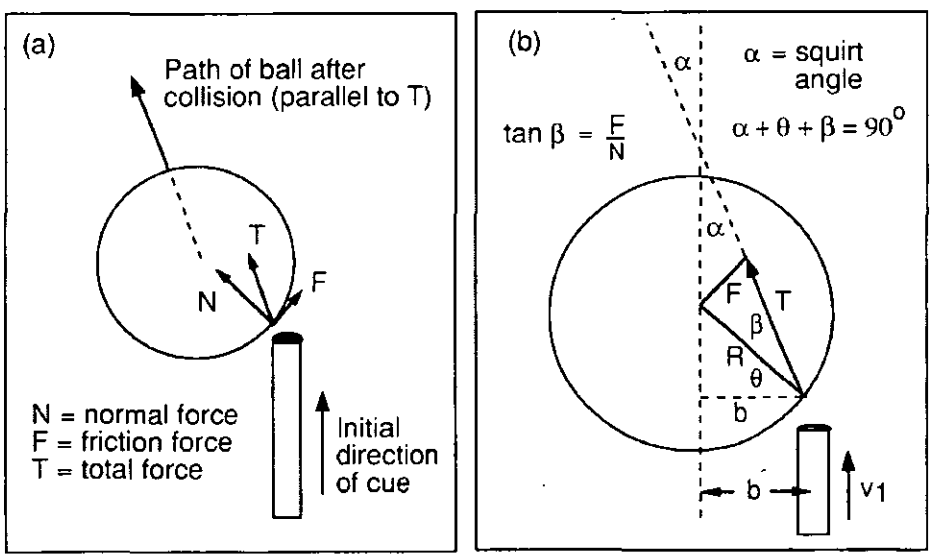


Fig. 2.11.The cue ball squirt **[Cross 2008]**

Cross **[2008]** has formulated a concise theory incorporating two important parameters that were not considered in the work of Shepard **[1997]**, namely, the friction coefficient and the coefficient of restitution between the cue and the ball. Moreover, Cross has experimented on the squirt phenomena using a pendulum-suspended cue ball, to eliminate the surface friction effects from the table. A video camera operating at 25 fps was used to track the cue and the ball to obtain their linear and angular velocities. The ball was marked with a line around its circumference that allows the spin measurement to be made from the camera (it should be noted that the pendulum suspended ball could only have sidespin). The theoretical prediction and the experimental values obtained by Cross are given in Figures 2.12 (a) and (b) respectively for the conditions of a chalked tip, an un-chalked tip, and a P800 emery paper attached to the cue ball. However, it must be noted that the values obtained by Cross cannot be used in a billiard robot as the results are not obtained for the situation where the cue ball rests on the table. The friction effects from the table are generally expected to influence the amount of cue squirt.

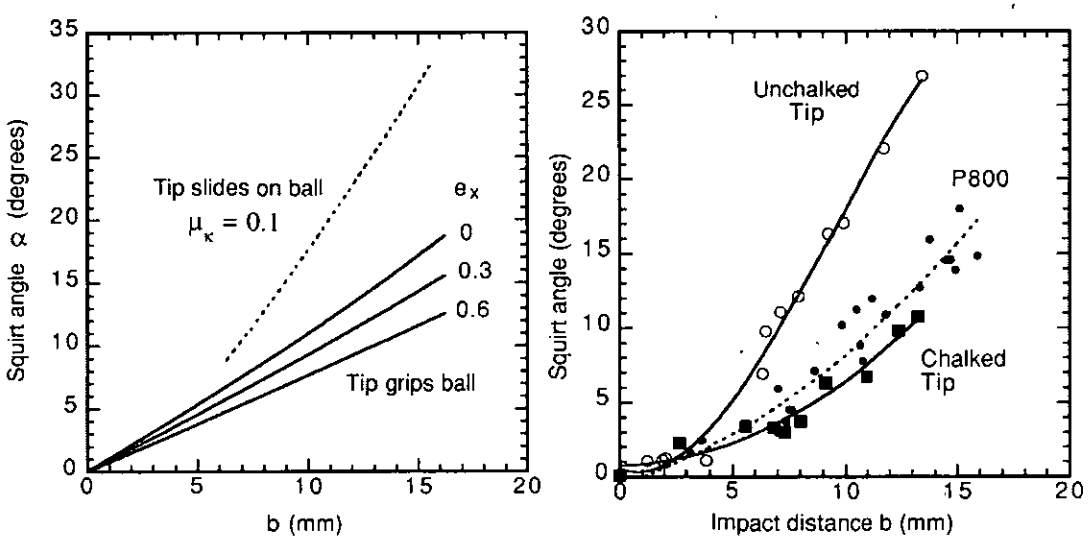
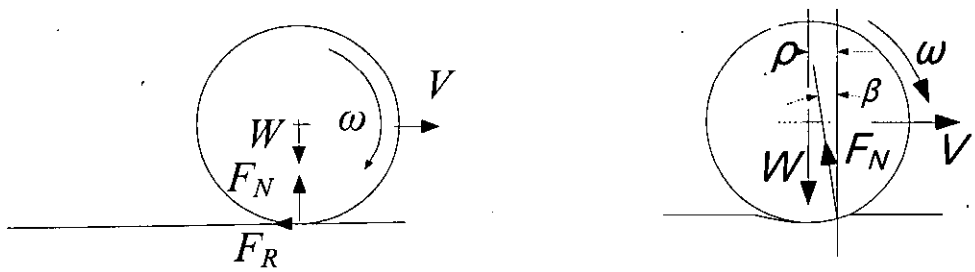


Fig. 2.12 Theoretical predictions and experimentally obtained values of cue ball squirt **[Cross 2008]**

2.4.2 Ball Motion against Friction on the Table

Once the ball starts its movement on the table, it either rolls on the table or slides. Hence, there are two friction coefficients to be taken into consideration. In order to

understand the dynamics of rolling and sliding, a snooker ball that has both linear, and angular velocities after cueing is depicted in Figure 2.13(a). No consideration is given to the sidespin of the ball as it is assumed not to affect the motion. This is known as decoupled motion, where sidespin is considered not to affect the linear velocity, V or the topspin, ω, of the ball and *vice versa*.



(a) (b)

Fig. 2.13. Forces acting on a moving ball

Immediately after cueing the ball, generally, it starts to slip on the table and this introduces a friction force, F_R , at the ball-table contact point, where $F_R = \mu F_N$. . But whenever either the spherical body (the ball) or the surface is deformable (the table cloth), the normal force F_N, from the surface does not go through the centre of the sphere, see Figure 2.13(b) **[Hierrezuelo and Carnero 1995].** In this case the horizontal component of the force F_N, $F_N \sin\beta$ acts against the motion, playing a role similar to that of the fiction force. Moreover, the action of F_N also introduces a torque in a direction opposite to that of the direction of rotation of the ball. Hence, an angular deceleration is also present in the ball. The relationship $V = R\omega$ is maintained throughout the rolling phase. An equivalent friction coefficient of μ_r can be defined for the rolling condition of the ball. μ_r is considered dependent on the nature and state of the surfaces in contact but not on the radius or the velocity of the sphere **[Domenech *et al.* 1987]**. It represents an arm of the pair of forces applied on the sphere perpendicular to the horizontal plane (Figure 2.13(b)).

Professional snooker players, by varying the point of strike on the ball, accurately control V_0 and ω_0 (initial velocities) in order to bring the ball to a desired location on the table. For example by imparting more top spin to the ball, the slipping phase can be made to stop sooner, and when bottom spin is given, the slipping will last longer.

This gives much more flexibility to the player rather than simply varying the linear velocity by a stun shot. Moreover, the amount of spin on the ball determines the collision dynamics (either with another ball or a cushion) and this aspect also gives the player a wide range of options.

The linear deceleration during the rolling phase is given by

$$\dot{V} = -\mu_r g$$

The linear velocity of the ball is written in terms of its initial rolling velocity V_0 as,

$$V = V_0 - \mu_r g t$$

Its angular velocity is given by, $\omega = \frac{V}{R} = \frac{V_0 - \mu_r g t}{R}$

The sliding friction coefficient, μ_s, is usually considered a constant according to Coulomb's law. But Witters and Duymelinck **[1986]** have obtained, using a simple apparatus consisting of a billiard ball and table cloth, a plot for the variation of μ with ball velocity, where it is observed that it increases with ball velocity asymptotically (but not smoothly) to 0.21, starting from 0.14. Gratton and Defrancesco **[2006]** experimented with sliding bodies on a flat surface and concluded that, whenever the velocity does not vary by more than a factor of two, μ_s is proportional to the sliding velocity. In addition, they also find a logarithmic relationship between μ_s and the sliding velocity. Some other analytical methods were also used, by analysing the deformation of material with few assumptions regarding the deformations, to calculate a theoretical value for μ_r **[Witters and Duymelinck 1986, Hierrezuelo and Carnero 1995]**.

When $V - R\omega > 0$, the ball slides on the table.

The linear deceleration of the ball due to sliding is,

$$\dot{V} = -\mu_s g$$

The ball will have an angular acceleration,

$\dot{\omega} = \frac{\mu_s MRg}{I}$, where I is the 2nd moment of inertia of the ball about any axis that goes through its centre, and $I = \frac{2}{5}MR^2$. Hence,

$$\dot{\omega} = \frac{5\mu_s g}{2R}$$

When, $V - R\omega < 0$, which is called 'overspinning', the friction force will be in the same direction as velocity V. This often happens when the cue impinges on the cue ball at a height larger than $\frac{7}{5}R$ from the table surface, or right after it impinges on an object ball (this effect will be seen later in the chapter). Hence, during the 'overspinning' phase, the ball will actually be accelerating and the value of the acceleration is given by, $\dot{V} = \mu_s g$

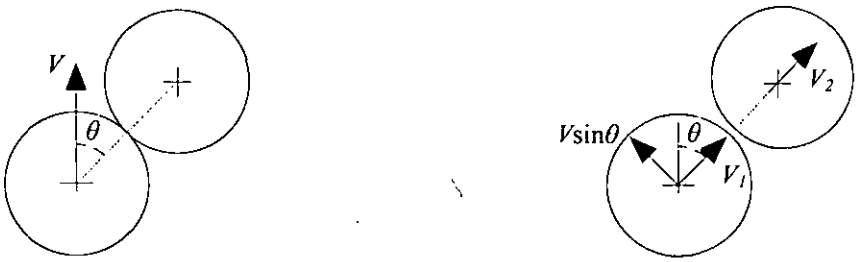
At the same time, its angular motion will be decelerating. The angular deceleration is,

$$\dot{\omega} = \frac{5\mu_s g}{2R}$$

Earlier, at the start of this section, the assumption of decoupled motion was stated (i.e. the sidespin does not affect the linear motion). However, as seen in the review of rolling friction, the ball always 'sinks' into the cloth and thus makes an area contact with the table. This clearly indicates that the point contact hypothesis that leads to the above assumption is usually incorrect. However, no physical analysis is found anywhere regarding this. If a significant amount of contact is made, the ball might have properties like that of a disk, which displays some interesting properties in its coupled linear-rotational motion **[Voyenli and Eriksen 1985]**.

2.4.3 Collision between Two Balls

Impact between two balls can be either frontal (head-on) or oblique. If the approaching and separating velocities lie along the line connecting the centres of the balls, then the impact is said to be frontal or head-on. Two-dimensional impacts are a general case in billiards and snooker and these are called oblique collisions, with frontal impact being a special case.



(a) Before colliding (b) After collision

Fig. 2.14. Pre-and post-collision velocities in plan view

Referring to the Figure 2.14,

Conservation of momentum, $V_1 + V_2 = V\cos\theta$

Coefficient of restitution relation, $V_2 - V_1 = eV\cos\theta$

Using the two equations, V_1 and V_2 can be easily calculated. The assumption is that both the balls have the same mass and the coefficient of restitution is e. According to this hypothesis, both the balls should move on linear paths after the collision. However, this does not happen in reality as is seen below.

90° Rule

Amateur snooker and pool players use the 90° rule to predict the cue ball path after the collision. It states that when the cue ball strikes an object ball with no topspin or bottom spin the two balls will always separate at 90° **[Alciatore 2004]**. It is evident from the two equations above that this rule is valid only when e=1, i.e. when the collision is perfectly elastic. Bayes and Scott **[1962]** employed a spring-loaded cue launcher and pool balls on a felt-covered table to examine this effect. A stroboscope and a camera were used to determine the subsequent ball paths and the experimental results showed that the separation angle was around 67° (though there is no evidence as to the amount of spin the ball had at the time of impact, as ballspin is known to affect the collision). Bayes and Scott also tested the ball on various glass surfaces such as dry, wet and soapy, and found that the ball separation angle approaches 90° as the surface gets smoother (in soapy glass it reached 89.9°). In the light of this evidence, it can be said the surface friction from the table leads to unpredictable rebound at impact.

Apart from the table friction, the ball rolling also works against the 90° rule. The friction between colliding balls plays a part in the motion when the cue ball is rolling. Domenech and Casasus **[1991]** considered the sliding friction effect between the balls during the collision, hence the friction impulse, and obtained the post-collision linear and angular velocities accordingly.

2.4.4 The Cushion-Ball Impact

When the ball does not have any spin, most literature assumes that the incoming ball angle and the reflected angle are equal ($\psi_i=\psi_o$) (see Figure 2.15 for the parameters involved in such a collision). Alciatore **[2004]** says that when the approaching velocity, *Vi*, is high, ψ_o tends to be larger than ψ_i, because of the elastic forces set up due to high deflection of the rail (cushion), this phenomenon is called 'throwback'. However, it must be noted that ψ_i directly influences throwback. In addition, the coefficient of restitution is less than unity and the lateral friction, along the cushion affects the impact thereby causing the change in the deflected angle ψ_o.

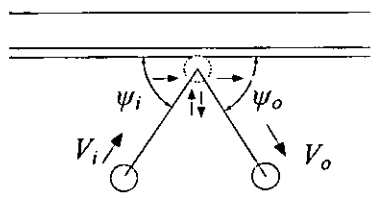


Fig. 2.15. The cushion-ball impact - the plan view

Partridge and Spong **[2000]** have considered the effect of friction on the puck-table impact for their air-hockey-playing robot. They use the Routh impact model, which is based on the Poisson restitution law and is more accurate, instead of the Newtonian restitution law. However, they only use sidespin in their analysis (because it is a 2-D motion situation and does not have any vertical velocities).

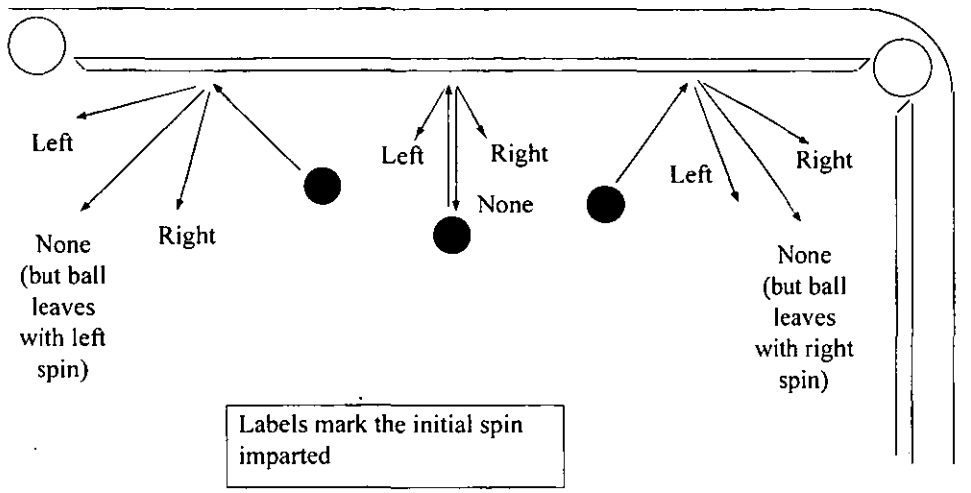


Fig. 2.16. Effects of putting spin on the cue ball on the collisions with the rail

The analysis for the snooker wall collisions has to include the top and bottom spin of the ball and, as a result, a vertical friction force will be present at the ball-cushion interface. A complete analysis should also incorporate the friction between the ball and table, as in the case of cueing. If the ball approaches the cushion with no English (sidespin) at an angle other than 90°, friction from the cushion gives it English. Figure 2.16 shows how the balls launched with different sidespins, at different angles to a cushion, change their paths.

Summary

This chapter provides a review of the literature found in relation to the development of a snooker robot. Strategies needed for the robot are reviewed both in the context of strategy games such as chess and from the perspective of snooker/billiards robots and related computer games. Machine-vision-related issues for a snooker robot are compared by reviewing what has been implemented in the existing robots. In addition, after identifying that ball tracking is important for a snooker system, related developments have been outlined. Moreover, finding that manipulation methodologies have not been established for such robots, a related robotic manipulation method called nonprehensile manipulation has been reviewed. Finally, a comprehensive review of the physics of snooker is given.

Chapter 3

RESEARCH METHODOLOGY AND PROPOSED SYSTEM OVERVIEW

The literature review chapter shows that robotic snooker/billiards spans different technical disciplines and this aspect underlines the need to adopt an interdisciplinary approach to make a successful system. This chapter tries to identify the critical gaps in the literature and sets appropriate research targets.

Section 3.1 identifies the problem to be addressed and describes the research methodology to be used. Section 3.2 details the hardware features needed for the proposed system. The controls and the associated software of the system are also outlined.

3.1 Research Methodology

3.1.1 Problem: Robotic Snooker

It is apparent that the artificial system should be able to plan and execute shots like a human in order to win a game of snooker. The success of the overall system is decided by the individual performances of each of the major elements of which it is made up. For a given system, generally subsystems can be identified in a multitude of ways, depending on the purpose of the differentiation procedure. Here, the system is broken down into three major subsystems **A**, **B**, and **C**, as described below, to differentiate and identify the research goals and the targets of this work.

Subsystem A: Shot identification

For the current table configuration, this selects the best shot considering the various factors outlined in Section 2.1, and selects the object ball-pocket

combination for potting and decides the best place to leave the cue ball after the shot.

Research Area: Artificial Intelligence.

Subsystem B: Element that plans the shot

Working out the inverse solutions for the shot that is identified by **A**. In order to achieve the targets set by **A**, **B** must make decisions regarding the launching direction of the cue, the cue velocity and the striking point on the cue ball. Therefore, **B** must have the knowledge of ball trajectories and the phenomena associated with the ball motion, such as rolling, sliding, spinning and impact mechanics.

Research Areas: Mechatronics, Dynamics, Robotic nonprehensile manipulation, and Machine learning.

Subsystem C: Shot execution

This element ensures that the shot that is planned by **B** is executed accordingly. It mainly consists of the design of a suitable hardware and software configuration by the proper identification of the system requirements.

Research Areas: Mechatronic system design, by appropriately integrating different sensors (also vision) and actuators with microcontroller and PC-based control, Robot inverse kinematics, and error calibration methods for the robots.

3.1.2 Part of the problem that is addressed by this thesis

After an extensive literature survey it was identified that issues that are related to **Subsystem A** have been extensively treated by computer scientists, and several game-playing programs have been created. However, these programs have not been fully implemented on real snooker/billiard robots to test their effectiveness and to compare them with human performance since the other two elements (i.e. **B** and **C**) of the overall solution have not yet been developed to perform on a par with humans.

This thesis does **not** concentrate on the research issues related to **Subsystem A.** Therefore, throughout this work, it is always assumed that for a given state of the

table, the best object ball-pocket combination and the preferred post-shot final cue ball location have always been supplied (i.e. they are always assumed).

The literature review also showed that there scarcely exists any literature on the inverse manipulation problem, which is the concern of **Subsystem B**, and this problem remains largely unsolved for snooker dynamics. Researchers interested in billiard physics have developed some dynamic models for impacts in snooker/billiards, but these models mostly remain incomplete. In addition, the parameters that affect different phases of the ball motion, such as rolling and spinning have not been properly measured. These parameter measurements are believed to require an extensive use of machine-vision-based, non-intrusive, experimental techniques. All-inclusive inverse trajectory solutions that are necessary for a problem like snooker have not been addressed by any of the researchers. Hence, there are multiple issues involved with **Subsystem B** such as computer vision, dynamics and robotic manipulation methodologies. The challenges related to **Subsystem B** will be the major focus of this work.

There are two full-fledged gantry-based robots to play snooker and pool **[Ho *et al.* 2007, Greenspan *et al.* 2008]**. Both robots were kinematically versatile and have the ability to reach any part of the table. However, the researchers in those respective projects have not tried to address certain issues such as accurately positioning the cue to play different spins on the cue ball. In the case of the Queen's University project, the robot is designed to have the facilities needed for visual servoing using a local camera mounted close to the cue **[Lam 2008]**, but imparting spin to the ball was not reported in their research papers. In order to test the performance of the methodologies that it is proposed to develop for **Subsystem B,** adequate hardware support should be available. Otherwise, only computer simulation-based results can be reported for the solutions proposed for **Subsystem B**. To test the performances of the proposed trajectory algorithms, a suitable and desirable system configuration is identified in Section 3.2. This, in turn, makes some valuable contributions towards the development of an ideal **Subsystem C**. Therefore, issues related to some aspects of **Subsystem C** will be a second major focus of this thesis.

3.1.3 Solution Methods

Nonprehensile methodologies for object trajectories require a description of the object's trajectory, here the ball trajectory. In order to determine the ball trajectories, Chapter 5 addresses the issue of measuring the physical parameters that are involved in the ball motion. Chapter 6 describes impact mechanics and numerical modelling-based solutions for snooker collisions, which drastically change the ball trajectories on the table. Chapter 7 proposes solutions based on what is known from the existing literature and the work of Chapters 5 and 6. An optimisation-based solution with a search procedure using Genetic Algorithms is investigated as a possible solution.

3.2 Proposed System Overview

The proposed system to play snooker (i.e. to test and validate the research goals) is based on the integration of sensors and actuators and is based on a mechatronic approach to system conceptualisation and realisation. Based on this, a synergistic design of the system, integrating the mechanics, electronic sensors and actuators with a PC, is envisaged. The shortcomings of the existing snooker/billiards systems (see Chapter 2 for a detailed survey), and their relative virtues were taken into account in order to offer a better system configuration. Some innovative ideas like force sensing during the cue-cue ball impact are put forward.

3.2.1 Hardware

Snooker is a game that involves very subtle human skills like accurate positioning and striking of the ball with very good control. Human limbs have superior agility and very good coordination and form the basis of these skills. Professional players master this game after thousands of hours of practice. Visual observation plays a major part in this. The human limbs are also superior in that they gather a feel of the force and impulse transfer during a shot and this *a priori* knowledge is used, unwittingly, for a given shot. All these capabilities warrant a careful system identification procedure of the hardware configuration for an artificial system to play snooker.

3.2.1.1 Precise Positioning and Controlled Manipulation of the Cue

As seen in Figure 2.6, the hitting point of the cue on the ball alters the type of spin imparted to the ball, greatly affecting the subsequent ball behaviour. Hence, a positioning arrangement for the cue is necessary, so that it can hit the cue ball at different spots as required. An accuracy of 1 mm is proposed for the positioning system that positions the cue on the ball. Humans must find it difficult to go beyond this accuracy on placing the cue over the cue ball. This is considering the fact that the white cue ball is without any guide markings showing the distance measures, and also that the player's eyes are generally at a distance of over a metre from the ball when taking aim.

Even though the existing systems are designed to occupy the whole workspace of the snooker/pool table, to play all possible object ball-pocket combinations, their manipulators have not been used to place the cue stick on the ball very accurately. One reason for that is that the spin shots were not targeted in their game strategy. Moreover, because of the long, serial-type manipulators used, positional errors are bound to propagate and the accurate positioning is difficult unless a very good error calibration and compensation is carried out for the robot. In this project, an accurate X-Y positioning system is envisaged to position the cue on the ball.

Since it is planned to use a regular unmodified cue in this project, imitating the human cueing by incorporating a cue bridge is very important. In snooker, the purpose of the cue bridge is to provide a guide through which to send the long cue on a straight line. Although when the cue strikes the cue ball, the duration of impact is a tiny fraction of a second according to Marlow **[1994]**, the absence of a proper reinforcement, by the way of a cue bridge, can bend the cue at its tip. When the cue bends about its tip, the follow-through of the shot will not be smooth. At this juncture, it is important to note the significance of the follow-through in sports where impacts are involved (e.g. cricket, tennis, baseball). These factors underscore the need for a cue bridge as an essential part of the cueing process.

The shot duration (the time from the start of the shot to until the time that the cue strikes the ball) is between 0.2s to 0.5s for humans **[Alciatore 2008]**. The accurate and controlled cueing method that is selected must also be fast enough to produce the

required cue velocities as found in humans. The pneumatic cylinder-driven cue launchers have poorly performed in the previously developed systems. Although Shu **[1994]** used the force control option in the pneumatic-driven cue manipulator, pneumatic systems are considered unreliable when it comes to the controllability of velocity. The pneumatic actuators will also hinder the rapid launching of the cue needed for very fast shots. However, humans launch the cue at very high velocities. The cue velocity even for a moderate shot is measured from Alciatore's **[2008]** high-speed video clips at around 2m/s. Hence, a motor-driven cue launcher is proposed.

3.2.1.2 Vision and Ball Tracking

The current systems playing snooker/pool use machine vision cameras only to locate the static ball positions on the table, in order to decide the best shot available and then to allow the robot to reach for the current cue ball location in order to strike it. Except for a project at MIT, where some efforts have been made to track the ball in order to demonstrate the effectiveness of a certain method of machine learning, no attempts have been made to track the ball continuously **[Moore 1991, Moore *et al.* 1995]**. However, visual tracking of the balls allows the system to observe the results for a given shot parameter set, and these tracking results can be used to determine what is actually taking place on the table. For instance, some physical parameters can be empirically derived from the analysis of the tracking results, and in some cases, a look-up-table-based method based on previous observations may even prove useful. In summary, the tracking of balls is an essential feature, given the nature of the game. The vision system, as described later in this chapter, is proposed to achieve this target.

It is proposed to mount the camera on the ceiling above the table, looking vertically down. As the ball motion is confined to a plane (i.e. the snooker table), a monocular camera is sufficient to determine the position of a ball on the table, given that the camera is calibrated and the spatial transformation between the camera plane and the table plane is known.

3.2.1.3 Other Sensors used

Vision is the primary sensing element of the system, as described earlier. However, additional sensory information is also required if higher-level reasoning about the system dynamics is required. This necessitates certain additional sensory information.

It is proposed to install a force sensor on the cue to get the tactile information about the cueing process. This force sensor is expected to be an equivalent of the tactile feel in humans. For example, the tactile feel distinguishes the shots wherein the cue is almost jabbed against the cue ball, as in the case of massé shots, where the cue is made to stop abruptly, against the smooth shots where the cue follows the cue ball closely even after the impulse has taken place, very smoothly, giving a better positional accuracy. In addition, when miscuing occurs (where adequate contact between the cue and the ball does not happen, the cue almost slips on the cue ball surface, which misguides the ball) there is not enough normal force transferred to the ball at the point of contact. These phenomena underscore the need for force-sensing and justify the inclusion of a force sensor. Importantly, the force sensor thus fitted to measure the force transfer from the cue to the cue ball should not alter/change the properties of the cue.

3.2.1.4 Snooker table

Fig. 3.1. The snooker table in the mechatronics lab

A Riley® Renaissance Type Snooker table having dimensions of 10ft x 5ft, has been installed in the lab (see Figure 3.1). This brand is the official table of the World Snooker association and is used for all its professional and amateur Snooker

tournaments since 1992 (except in China) **[WPBSA 2008]**. The regular tournament table size is 12ft x 6ft.

3.2.2 Controls and Software

The system is not intended to operate in real-time. The nature of the game allows for a certain time gap between successive shots, and thus the constraint of operating in real-time is not strictly imposed on the proposed system. Visual Basic® 6.0 (VB) is used as the programming language for the main control program.

Image processing and analysis are to be done within MATLAB®. This is in view of the availability of a wide range of built-in functions for the image analysis in the MATLAB® Image Processing Toolbox. The vision algorithms are to be written as M-files in MATLAB®. These M-files are then to be called from VB using a function procedure called MATLAB® COM component, which is generally used to integrate MATLAB® with other programming environments.

It is proposed to control all other hardware from within VB.

Summary

This chapter identifies the research objectives of the project. The proposed research methodology to carry out the research is summarised as well. The envisaged hardware requirements of the system required in order to meet the research objectives are also given. The controls of the system from a PC are also briefly outlined.

Chapter 4

DESIGN AND REALISATION OF THE ROBOTIC SYSTEM

This section describes how the system is realised as per the requirements set out in the previous chapter. The tools and techniques used are explained and the final outcomes of these processes are presented as drawings and data. The first section explains the requirements and considerations that are taken into account to determine the unit for the cue manipulation. In Section 4.2, the cue positioning unit is described with appropriate drawings. Section 4.3 addresses the issue of machine vision, image processing, related issues, and the problems encountered in each of these aspects.

4.1 The Cue Launcher

The cue launcher has to achieve the stroke-velocity-acceleration requirements as needed for any given shot. Various considerations regarding the design of the cue launcher configuration are now described in detail.

4.1.1 Some Considerations regarding the Snooker Cue

Automated cueing operation is vital for the overall performance of the robot to be satisfactory. Although the Bristol University project had a robot with a large workspace, it used a pneumatic cylinder fitted with a cue tip of a regular snooker cue **[Shu 1994]**. The front half of the cue was used for the Queen's Ontario pool playing robot **[Long *et al.* 2004]**. Other researchers have used various forms of rounded heads replacing the regular billiard cue completely **[Alian *et al.* 2004]**. However, in reality, it is believed that the cue in its full form, shape and mass distribution plays many different roles during the shot-making process. Figure 4.1 shows the picture of a typical snooker cue.

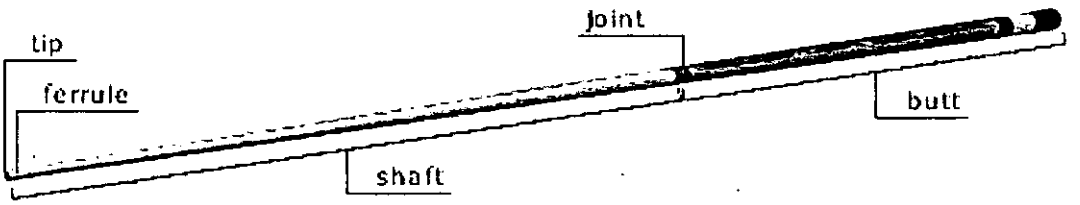


Fig. 4.1. A snooker cue

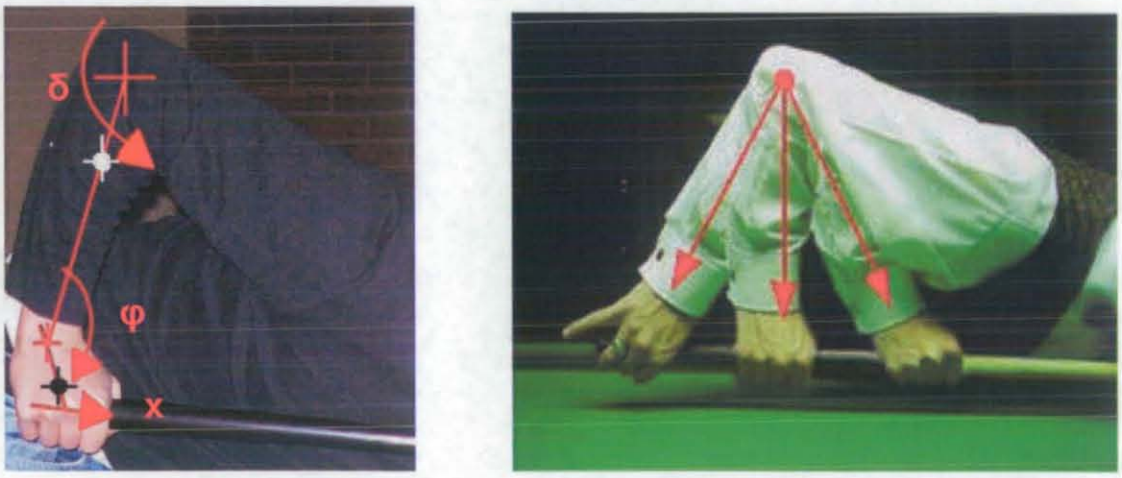
4.1.1.1 Cue Mass

A snooker cue usually weighs around 500g (18-21 oz). If the cue is very lightweight, significantly lower than the cue ball mass, even if it is manipulated with very high velocity the momentum and energy transfer to the ball will not be effective **[Marlow 1994]**. In addition, a low cue mass generally implies a reduced stiffness. Since smooth follow-through is very important to cue sports, this results in lateral cue deflection during the impact and produces miscued shots **[Williams 2002]**. Moreover, to achieve a specific ball velocity, a particular amount of momentum has to be transferred by the cue to the ball. Now, with a reduced cue mass, and to transfer the same momentum as before, the cue velocity has to be larger. However, the maximum possible velocity is limited by the prime mover, which is proposed to be a motor. In addition, the rotary to linear motion conversion limits the linear speeds if a rotary motor is to be used. Given the fact that linear motors operate at comparably lower linear speeds, the only available option is to go for a rotary motor, for which the problem of speed limitation due to the motion conversion arises. For example, the lead screws can only achieve maximum velocities of a few hundred millimetres per second. This may be a limiting factor and set a lower limit for the cue mass.

Conversely, if the cue and the associated mass are large, the velocity of the cue can be small whilst still having the same momentum as before, but the cue launcher will be heavy and its mobility will be difficult. In addition, a comparably large mass moving slowly cannot make the cue ball move faster after the impulse (the energy transfer will be very poor). Hence, there is a compromise between the mass and the associated velocity of the cue. For the maximum energy transfer from the cue to the ball, it can be proved, mathematically, that the cue should have the same mass as the cue ball. However, the cue mass is generally 3 times that of the ball mass because the focus

here is not on achieving the maximum energy transfer, but the controllability of the energy transfer **[Shepard 1997]**. The present author found it extremely difficult (or virtually impossible) to play fast shots with a cue where the rear half (butt) of the cue was removed; even for slow shots, a lack of stability was experienced by the arm. The design of the cue launcher strives to keep the inertia of the moving components as close as possible to the human cueing. The forearm essentially carries out cueing (see Figure 4.2). Hence, by using the typical inertial properties for this portion of the upper human limb, it is possible to get an estimate of the inertia involved in human cueing. This value must be considered when selecting the motor and the resulting inertia of the motorised configuration must be kept close to this value.

In order to calculate the equivalent mass involved in human cueing, a simple calculation is performed. As shown in Figures 4.2(a) and (b), the arm movement involves a rotation of the forearm about the elbow and a linear motion of the hand while having a rotation about the wrist. Although there is a configuration change in the hand to accomplish the desired straight-line motion of the cue, this change is neglected and its effect on the linear inertia is ignored. Hence, in Figure 4.2(a), φ is assumed to remain constant during the swing of the forearm, which is denoted by δ.



(a) (b)

Fig. 4.2. The arm movement during the stroke

Table 4.1. Properties of arm segments for a 6 ft tall male subject weighing 75 kg **[Clauser *et al.* 1969]**

Segment	Mass/ kg	Length/ m	Location of the centre of gravity/ m	Second moment of Inertia/ kgm^2
Forearm	1.21	0.254	0.109 (from the elbow)	0.0066 (about the frontal axis through COG)
Hand	0.46	0.181		

In addition, it is also assumed that the hand moves linearly with the cue. Now, using the concepts of linear and angular momentum in conjunction with the parallel axis theorem to calculate the second moment of inertia about different axes, and using the values from Table 4.1, the equivalent linear inertia involved, including the cue mass (i.e. 0.5 kg), in the cueing is derived as 1.13 kg.

4.1.1.2 The Shape and Material of the Cue

The World Snooker Association, the governing body for snooker, restricts the minimum cue length to 3 ft (0.914 m) also asserting that the design shall not depart from the traditional and generally-accepted shape and form **[WPBSA 2008]**. The cue taper aids in both easy holding for the human hands as well as providing good positioning abilities by being made slender at the tip. The material of the cue, i.e. wood with its very high damping characteristics, absorbs a great deal of energy that is generated due to the impact. The cue, also being very flexible, vibrates transversely during and immediately after the impact thus helping to absorb the energy further.

4.1.1.3 Cue Tip

The highly elastic cue tip (usually made out of leather or of synthetic materials combined with leather) is compressed during the collision between the cue and the cue ball. When the compressed cue tip recovers its shape the cue ball is pushed away from the cue. Its soft nature also ensures that it does not make a dent on the cue ball

surface. So, if metal parts replace the standard cue tip, as has been done in a few projects **[Alian *et al.* 2004]**, the cue ball may not travel as fast as with the regular tip, for a given cueing velocity.

4.1.2 Length of Stroke, Velocity, and Acceleration Requirements

This section aims to define some numerical values for the motion parameters of human cueing, in order to identify the requirements for the motor to be selected.

4.1.2.1 Length of Stroke

The length of stroke does not vary too much in snooker, because it is related to the player's comfort and skills. Once a player starts to alternate the length of stroke excessively, the 'feel' for the shot-making will be lost. Hence, some consistency can be assumed in the length of stroke. However, there are also other considerations, like the power of the shot, which plays a part in the stroke of the cue. No study exists in this regard. On making some measurements with different players (some reasonably good players, who can pot 5 balls continuously on regular basis, and amateurs) with a metre scale, and using some of the videos available from Alciatore **[2008]**, it is found that the stroke lengths generally vary between 120-250 mm.

4.1.2.2 Velocity

The required velocity of the cue depends on how powerful the shot has to be. The equivalent cue mass (cue mass and the associated inertia of the moving upper limb) depends on the nature of the grip of the hand. No study exists in this regard, and the effects of different grips have been neglected, hence, it is assumed that here that a tight grip exists. Hence, the equivalent mass of the cue side is treated as a constant. Now, also using the principle of momentum conservation, the cue ball velocity depends only on the cue velocity. In pool, the fastest cue ball speed reported ever is 15.6 m/s, played by a martial arts student, and more typical break speeds are around 9 m/s **[Shepard 1997]**. However, in snooker, the shots are relatively slower when compared with pool. Since a permanent machine vision camera for the system is not available already, some experiments to measure the cue velocity have been performed by using a NAC HSV-400 high-speed video camera. A calibrated scale is placed

behind the cue ball, for distance measurements along the horizontal, and cueing is tracked at 1,000 frames per second (fps). For very fast shots (similar to break shots), the cue velocity is found to be in the range of 3-4 m/s. The cue velocity from the proposed motor drive is expected to attain velocities of this magnitude.

4.1.2.3 Acceleration

Since the cue is expected to reach velocities of around 4 m/s, the cueing involves very high accelerations within a short duration. However, the rules of the game suggest that there has to be a single impulse between the cue and the cue ball, i.e. multiple impacts are prohibited by the rules of the game **[WPBSA 2008]**. This rule constricts the cue motion in that there cannot be any acceleration on the cue immediately after it has struck the cue ball. It must be noted here that the cue ball will decelerate immediately after the impact due to the friction from the table. Hence, if inadequate consideration is given to the cue motion profile at the time of impact, multiple impacts may possibly happen. Therefore, it is safer to have zero acceleration when the cue strikes the cue ball, and this feature must be available in the cue launcher. For a maximum cue velocity of 4 m/s to be reached from rest, for example, within 0.2 s (typical cueing times are given in Section 3.2.1.1), the average cue acceleration has to be 20 m/s^2. The motor should be powerful enough torque to have this amount of acceleration, which produces high inertial forces in the cue launcher. Besides, the motor must also be able to accelerate at this rate.

4.1.3 Force Transferred to the Ball

When the cue strikes the ball, a large force is transferred within a very short time. The force calculations at the cue tip-cue ball interaction is important, as the selected motor must have adequate driving torque to counter this high force during impact. In addition, the contact force is needed to calculate the energy loss during cueing. The energy loss, in turn, will influence the power of the selected motor drive. Knowing the order of magnitude of the force is also important for the selection of a suitable force sensor for this application.

A review of the existing literature shows that the force values encountered in snooker/billiards cueing are not documented anywhere. Simple experiments are performed together with a few calculations to estimate the force at the cue-cue ball interface. During the impact of the cue on the ball, the force initially increases with time attaining a peak value and decreasing as the ball separates completely from the cue. Marlow **[1994]** suggests a sine-squared profile for this variation, as shown in Figure 4.3. To any general impact, the impact-momentum equation $\int F.dt = MV_G$ holds. Where M, the mass of the ball, is 142 g (the tolerance in the ball mass in a particular ball set is to be within 3 g according to WPBSA, the governing body of snooker).

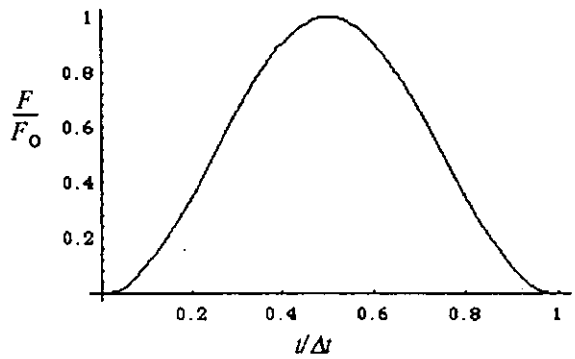


Fig. 17. Form of the sine squared time dependent force profile.

Fig. 4.3. A force vs. time prediction for the cue-ball impulse **[Marlow 1994]**, F_O denotes the peak force

The idea here is to calculate the contact force by estimating the momentum transferred to the ball together with the time taken for the collision between the cue and the cue ball. The literature consists of some information about the impact time variation with respect to other factors, such as the cue ball velocity. Marlow describes an experimental procedure to measure the impact time where the cue tip and the cue ball are separately wrapped with aluminium foil and using a capacitive-resistive circuit with a digital voltmeter **[Marlow 1994]**. The circuit is closed by the aluminium foils coming into contact during the time of impact, then, the charge that is accumulated in the capacitor is used to get a value for the impact time, which is the time the circuit is kept closed. A similar setup used by Marlow **[1994]** to measure the collision time

between two balls is shown in Figure 4.4. The resulting impulse time for the cue–cue ball impact is given as a plot against the cue ball velocity (see Figure 4.5).

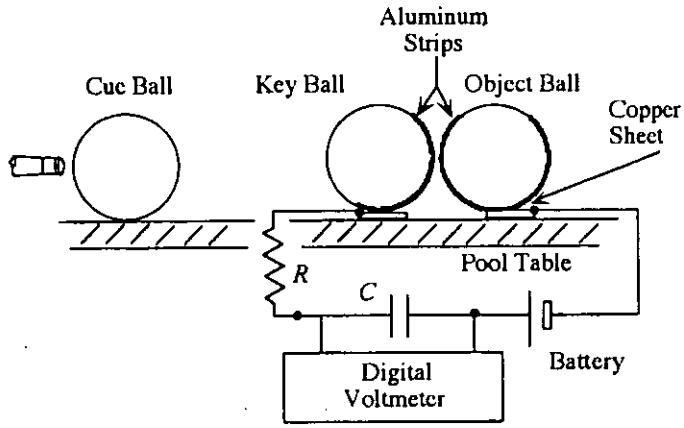


Fig. 18. The experimental set up to measure collision time between two billiard balls. The capacitance used was 10 μF and two values of resistance were used, 1000 Ω and 500 Ω.

Fig. 4.4. Measurement of collision time between two pool balls **[Marlow 1994]**

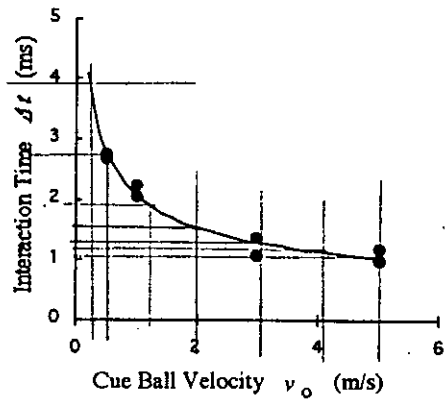


Fig. 89. The experimental data (eight points) for measurements of the interaction time as a function of the Cue Ball velocity. The curve is a least squares fit to a $v_0^{-1/3}$ functional of the interaction time on the incident Cue Ball velocity.

Fig. 4.5. Time taken for the impulse vs. cue ball velocity **[Marlow 1994]**

The data available from Figure 4.5 are used in conjunction with David Alciatore's high-speed pool shot videos to determine the order of magnitude for the force involved **[Alciatore 2008]**. From a video captured at 1000 fps, approximate measurements are made manually to determine the ball velocities and impulse time (spatial measurements are made by comparing the image parameters to the size of the

ball in the image, its real size being known). For example, a 4 m/s velocity of the cue prior to impact results in a cue ball velocity of 6 m/s and is approximately found to have a 1 ms (associated with a ±0.5 ms tolerance) impulse time (based on visual estimation of the contact between the cue and ball). This time value closely agrees with the variation available in Figure 4.5 (a little above 1 ms). Now, for these experimental values, using the equation $\int F.dt = MV_G$ together with a triangular force function approximating the variation given in Figure 4.5, the peak force, *Fmax*, can be calculated.

$$0.5 Fmax.\ \Delta t = MV_G$$

$$Fmax = 0.142 \text{ x } 6 / (0.5 \text{ x } 1 \text{ x } 10^{-3})$$

$$= 1700 \text{ N}$$

Notably, 6 m/s is a fairly high cue ball speed and is considered a high-powered shot.

In addition, a purely experimental procedure involving striking the ball with an impulse hammer is also performed in order to obtain the force values and to confirm the values obtained from the high speed videos of Alciatore **[2008]**. A Bruel & Kjaer Impulse Hammer with a B&K Nexus Conditioning Amplifier is used to measure the force on hitting a snooker ball (see Figure 4.6).

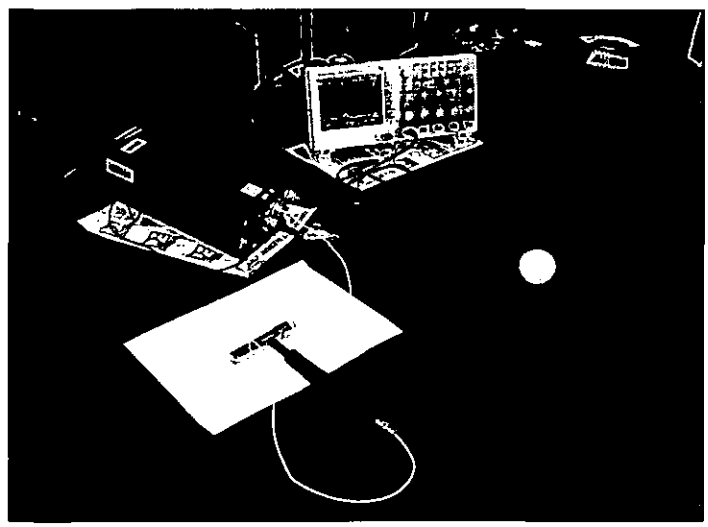


Fig. 4.6. Force measurements with an impact hammer

The ball velocity is measured approximately using a stopwatch and a tape measure. The waveform is observed in an oscilloscope to measure the maximum force and the

time taken for impulse. It is also observed that the force-time variation during the impulse is almost as described by Marlow **[1994]** and shown in Figure 4.3. High-speed shots are found not to be possible with the impulse hammer due to the difficulty in hitting the ball at its stun point at high speeds (a slight off centre hit is found to be difficult as well, since the impulse hammer tip does not have very good frictional properties). However, for a shot with 1.4 m/s cue ball speed (which is an average speed shot) the impulse time is measured as 2 ms and a peak force of 0.55 kN is obtained. Therefore the values obtained from two independent methods agree when it comes to the order of magnitude of the forces and the interaction times involved. These force values are considered for the selection of the motor.

4.1.4 Power Needed for Cueing

The calculation of cueing power is performed from the measurements made from David Alciatore's high-speed videos **[Alciatore 2008]**. As seen in Section 4.1.3, for a cue speed (V_c) of 4m/s the impact time is found to be 1ms and the resulting ball velocity (V_b) is 6 m/s. A peak force of 1.7 kN (corresponding to a triangular variation) or an average force of 0.85 kN is obtained when the impulse-momentum equation is used.

Cue power available, $\dot{E}_c = F.V_c$

$= 0.9 \times 1000 \times 4$

$= 3.6$ kW

Energy transferred from the cue $E_c = \dot{E}_c.\Delta t$

$= 3.6 \times 1000 \times 0.001$

$= 3.6$ J

The energy of the cue ball immediately after the impact

$E_b = \frac{1}{2}(M.V_b^2)$

$= 0.5 \times 0.142 \times (6)^2$

$= 2.6$ J

Energy efficiency of the cueing process $= 2.6/3.6$

$= 0.71$

This indicates an almost 30% energy loss at the cueing. When other losses, like the power loss due to friction at the linear guides (a linear guide is needed to guide the cue motion) and the losses in the motor gearbox, are also considered, the energy needed will well exceed the value of 3.6 J. For this particular shot, say the stroke is 10 cm (this is around the least possible region, as given in Section 4.1.2.1), and assume that the cue is at constant acceleration from the start of the stroke, the time needed to reach 4 m/s can be calculated in the following manner.

From, $v^2 = u^2 + 2as$, as the cue starts from rest, $a = \frac{v^2}{2s}$

$$a = 4^2/(2 \times 0.1)$$

$$= 80 \text{ m/s}^2$$

This is very high as a very small stroke length is assumed (this could lead to a smaller power-up time) to be on the safe side when calculating the motor power.

Now using, $s = ut + \frac{1}{2}at^2$, with u =0, $t = \sqrt{\frac{2s}{a}}$

$$t = (2 \times 0.1/80)^{0.5}$$

$$= 0.05 \text{ s}$$

Hence, the marginal energy of 3.6 J must be developed within 0.05 seconds needing a cue power of 72 W. However, the power rating of the selected motor must be higher than 72 W, as it also has to overcome various other losses, as outlined above.

4.1.5 Mechanical Manipulation of the Cue

Mechanical components should have an element to convert the rotational motion of the motor to a linear motion that drives the cue. Several options are considered. The human cueing operation is analogous to a linkage-based motion conversion. By having a two-element link and two motors to represent the elbow joint and the wrist joint, a linear motion can be obtained at one end of the linkage. A two-motor solution, although sophisticated, introduces many additional complexities such as: increased cost, weight, bulkiness, coordinated control of the two motors, etc, and thus is deemed unsuitable.

Other standard solutions such as worm and wheel, belt and pulley-based linear motion conversion, are deemed unsuitable considering the maximum speeds obtainable and the rigidity in manipulation, respectively. When discarding these alternative motion conversion methods, one of the obvious choices is a rack and pinion. This solution has the advantage of very high linear speeds that is suitable for the operation, rigidity and compactness - given adequately rigid support conditions. When compared to the linkage-based operation the linear cue displacement is also simply related to the motor rotation by, $x=k\theta$, where k is a constant.

4.1.5.1 Rack, Pinion and Slider

Factors such as the length of stroke, the allowable load of the rack (since it has to withstand high impact forces) and the mass involved should be in accordance with the mass considerations, as given in Section 4.1.1.1. For the impact forces under consideration, as defined in Section 4.1.3, it is not possible to select a rack that also complies with the mass requirement as outlined in Section 4.1.1.1. A rack from HPC Gears Ltd, with a length of 300 mm and a pitch of 2.0 module and a mass of 0.83 kg was selected. A pinion of 2.0 module, 30 mm diameter, and 0.1 kg mass is also chosen. The pinion will be coupled to the motor through a gearbox, and the rack has the cue attached underneath it by means of a bolted holder (there are provisions for two holders underneath, as shown in Figure 4.7, but one is considered to be sufficient). The rack is bolted to two rails at both sides, which is in turn sliding on 4 carriages, as shown in Figure 4.7. The carriages are rigidly attached to the main frame of the cueing device. The location of the carriages allows a maximum stroke of 170 mm. NSK carriages and rails are selected after checking for the allowable load for an assumed life span of the sliding system with the appropriate loading considerations as set out in the NSK product selection guide.

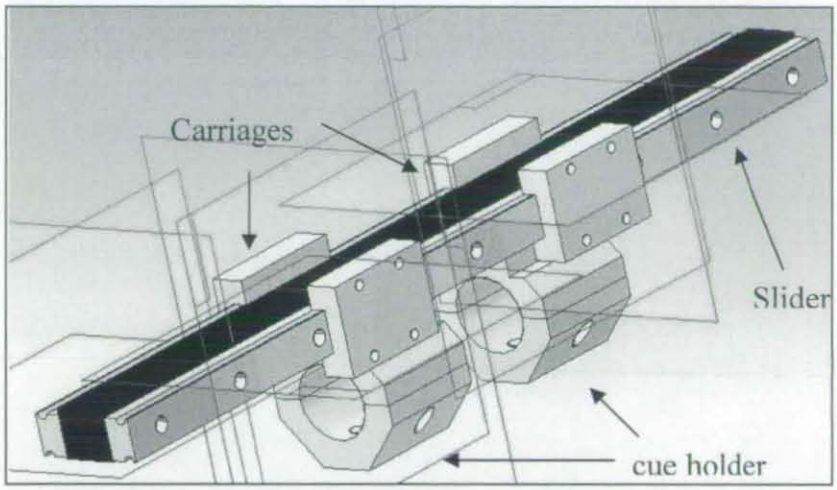


Fig. 4.7. The rack and slider with the cue holder bolted underneath

The cue can be attached to the cue holder blocks (see Figure 4.7), either rigidly, by directly clamping the cue between the aluminium cue holders, or 'softly', by introducing a rubber pad between the cue and the cue holders. The latter will closely resemble a human palm that dampens some of the impulse, but accuracy and repeatability issues may arise.

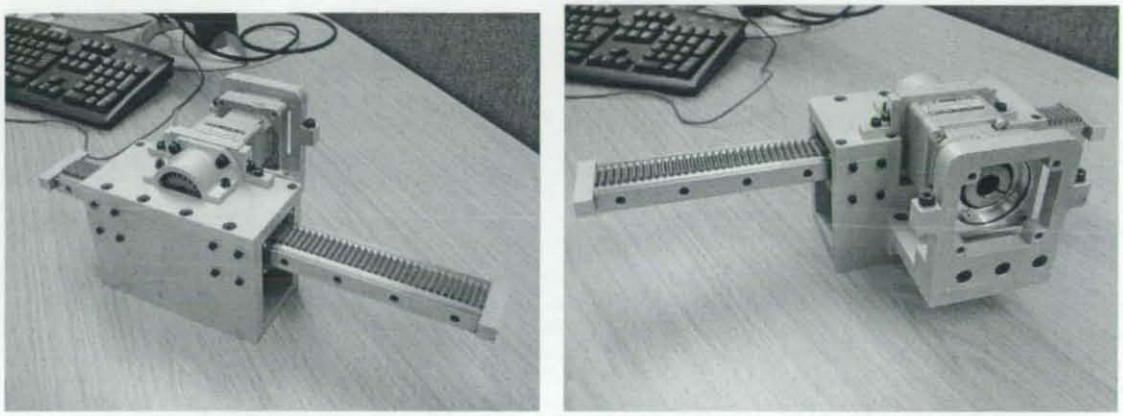
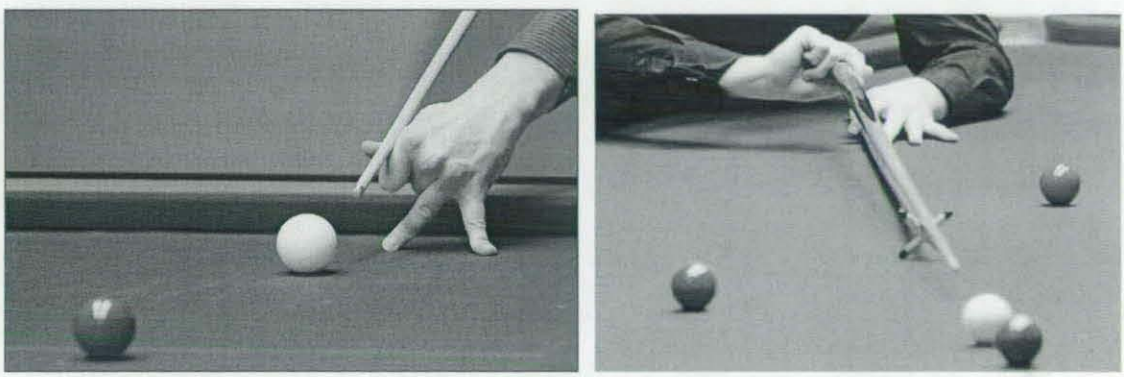


Fig. 4.8. Two views of the cue launcher- without motor

All the components of the cueing device are fabricated out of aluminium to reduce the weight of the structure and for aesthetic reasons. Bolted connections are used because of the alignment problems that can occur with the 4-carriage system during the assembly stage. Two photographs of the assembled cueing device are shown in Figure 4.8. The unit is 0.105 m x 0.160 m x 0.375 m in dimension (including the rack) and weighs 3.3 kg (without the motor)

4.1.6 The Cue Bridge

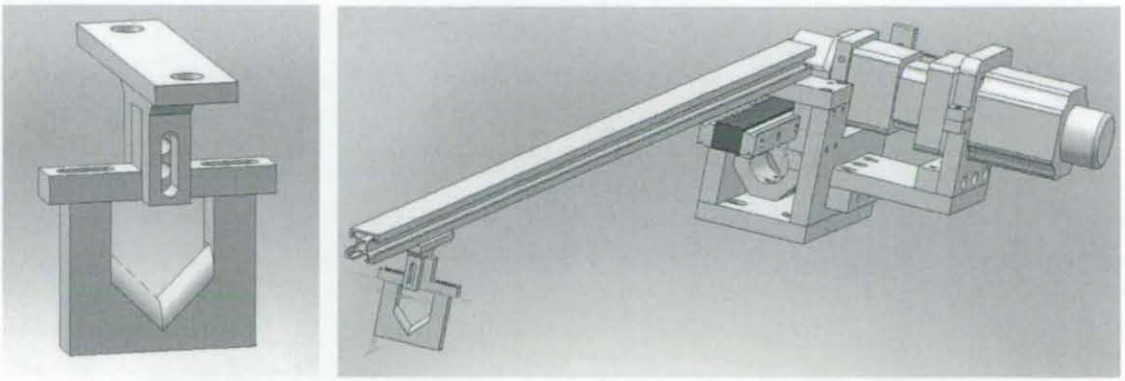
A cue bridge (see Figure 4.9 (a)) is very important from the human perspective of playing snooker. The figure shows a type of hand-bridge (there are also many other configurations used) and Figure 4.9 (b) shows a bridge that is used where it is difficult to form a hand bridge. The cue bridge helps positioning the cue on the ball accurately. In addition, it facilitates a smooth movement of the cue with less friction. Moreover, it helps to achieve a smooth follow-through.



(a) (b)

Fig. 4.9. Arm bridge and a wooden bridge

The bridge allows the cue to undergo transverse vibrations immediately after the impulse, suppressing the dynamic effects due to the impulse. A simple V bridge made of aluminium is used with the cueing device. As the cue is driven parallel to its axis and due to there being a slope of 0.375° in the cue, there is a 1 mm upward shift for a stroke length of 170 mm on the cue bridge, which is minimal. Alternatively, this can be taken care of suitably, by means of a vertical adjustment to the bridge. The configuration shown in Figure 4.10(a) is designed so that the bridge's position is adjustable horizontally and vertically up to 20 mm. The arrangement of the bridge attachment to the cue launcher is shown in Figure 4.10(b).



(a) (b)

Fig. 4.10. Cue bridge and its attachment to the cue launcher

A unit that supports the cue bridge and also rests on the table has also been designed to give stability to the front portion of the cue. Here, it must be noted that only the main body of the cue launcher (see Figure 4.10(b)) is given structural support and its front part resembles a cantilevered beam. This cantilevered configuration can give rise to vibrations and this may in turn change the point of impact with the ball thereby imparting a different spin to the ball to that required from the robotic system. A picture of the frontal support unit, which holds the overhanging part of the cue launcher, is shown in Figure 4.11. The jaws that hold the cue launcher can be manually adjusted either in the vertical or horizontal plane according to the motion of the platform on which the main body of the cue launcher is mounted.

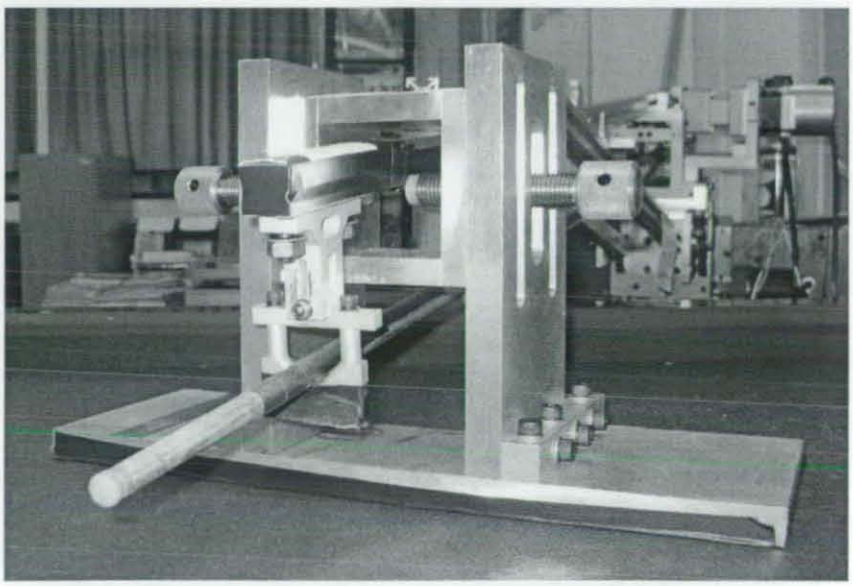


Fig. 4.11. Frontal support for the cue launcher

4.1.7 Drive Motor for the Cue

Both DC motors and servomotors are considered as potential candidates for the cue launcher unit. Considering the precise control needed for cueing, and the very accurate positioning abilities of servomotors, servos are deemed suitable for the cue manipulation operation. Both the brush type and brushless type servomotors, both DC and AC powered, were considered. After much consideration about the cost, type of power supply needed, encoder/tachometer availability, weight and also in accordance with the analysis performed in Section 4.1.4, a servo system from *SureServo*™ called the "200 W Low Inertia System" was selected. This uses a brushless DC motor. The drive amplifier also consists of a built-in power supply. The motor has been fixed with a 10,000ppr encoder enabling very accurate feedback control. Similar models from other manufacturers generally only have one tenth of this resolution. The equivalent linear inertia of the cue driving system is estimated to be 5.5 kg.

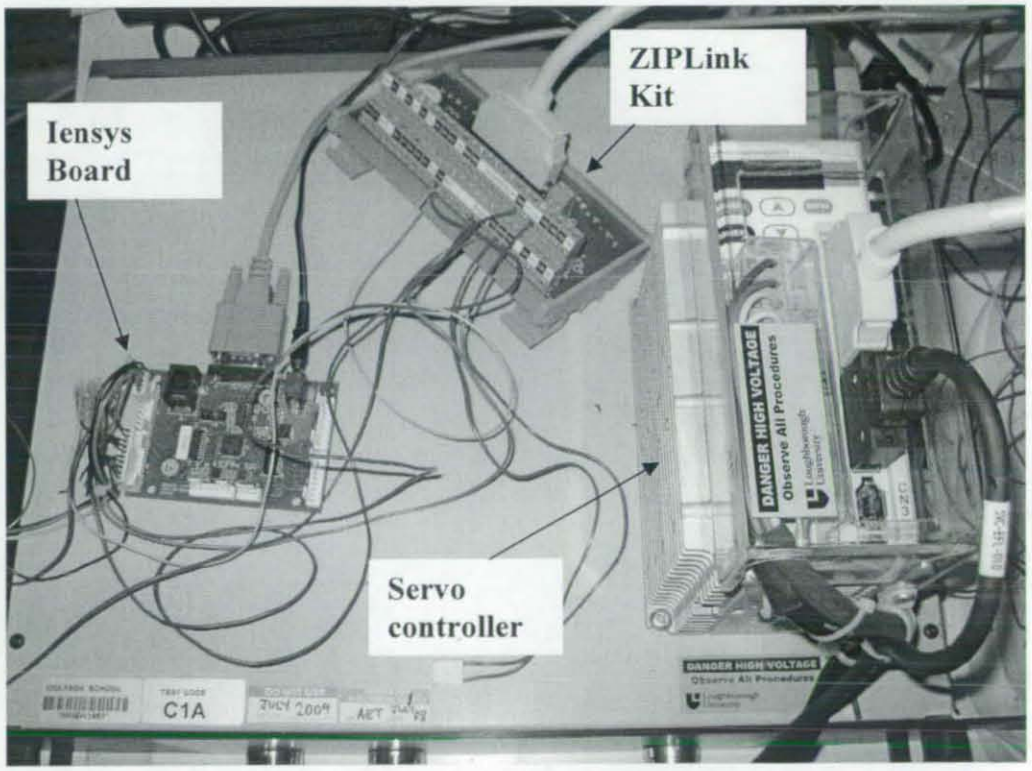


Fig. 4.12. External pulse control of the servo unit

Servo drive parameters can be programmed by using the servo drive's built-in keypad with LED display or through *SureServo* Pro® configuration software from within a computer. There are three control modes available: position, velocity and torque. The velocity and torque can be controlled with a ±10V analogue input signal or with the onboard Internal Indexer. The motor can also be controlled by the pulse and directional inputs from a PLC, a microcontroller or a line driver encoder. The controlling pulse rate can be as high as 500 kpps (kilo pulse per second).

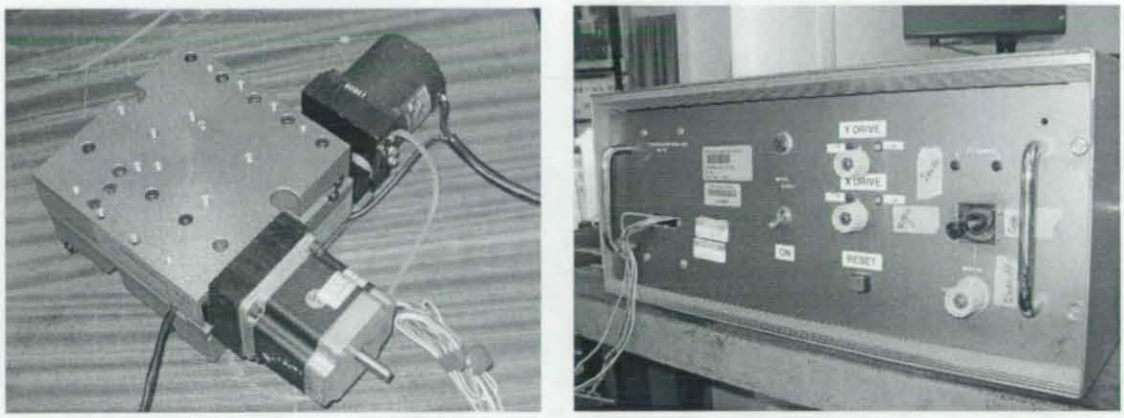
Here, the servomotor is controlled from an Iensys® board through a terminal block called the ZIPLink kit (see Figure 4.12). The Iensys® board's 4 output pins are programmed to emulate the 4 output channels of a quadrature encoder (see Appendix I for the connection diagram). The pulse rate decides the speed of the launched cue. The microcontroller decides on the pulse rate depending on a string that it receives from the PC via the RS-232 port. By changing the phase sequence of pulses, the motor is reversed at the end of a stroke to its original position. The servo controller also has an electronic gear ratio setting whereby the user can scale the high-velocity positioning pulses coming into the drive. Using the ZIPLink kit block, it is also possible to obtain the encoder readings of the servo, for monitoring purposes.

A 3:1 reduction gear box from Shimpo Drives® is fitted to the motor to increase the output torque at the load. The gearbox also ensures that the motor is operated at its rated speed.

4.2 Cue Positioning System

Accurate positioning of the cue on the ball is taken care of by a stepper motor stage. The stepper drive is an AEROTECH® ATS302 2-axis linear stage, as seen in Figure 4.13. This has been a spare unit in the mechatronics lab, and is not custom-selected for this application. It has a linear positioning accuracy of 2 μm/pulse, which is more than adequate for the positioning purposes. In 'Auto' mode, it is controlled by the PC through a DB-25 parallel port. It also has a 'Manual' mode, where a toggle switch is used to move the X and Y motors. Although it was initially programmed to be

operated from VB, since the drive itself does not have any encoder, the 'Manual' mode is expected to be used. Both axes have a maximum travel of 50 mm. The load capacity is 10 kg each for the axes. Since the vertical load from the cue launcher marginally exceeds this value, a new high torque motor has been fitted to the stage.



(a) (b)

Fig. 4.13. Two-axis AEROTECH® stepper drive and its controller

The stepper is mounted on the bottom of the cue launcher through a bracket fabricated out of aluminium (see Figure 4.14). The bracket has a slot by which cue tilt angle can be adjusted.

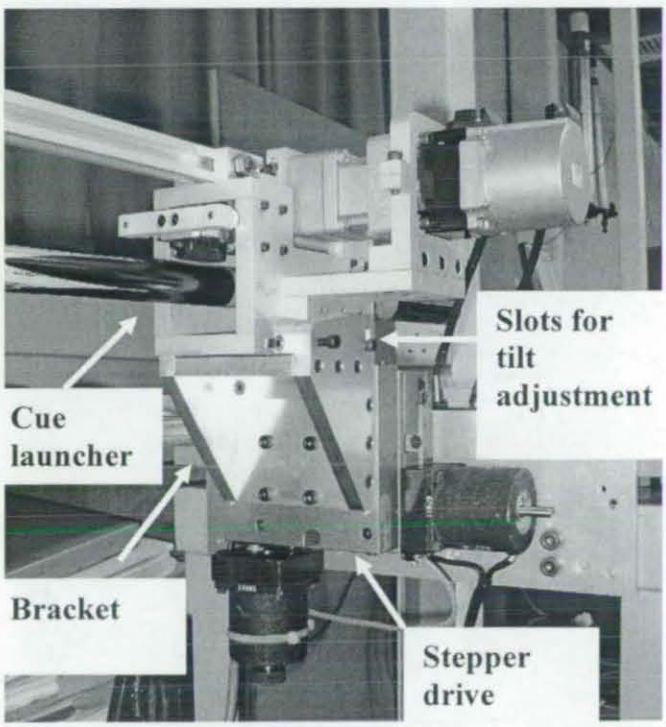


Fig.4.14. The cue launcher mounted on the stepper drive assembly

4.3 Camera and Vision

When selecting a camera, it is preferable to opt for a relatively high-resolution camera, to minimise any compromise on the spatial accuracy issues. The snooker table dimensions are 3x1.5m and a spatial resolution of around 1mm/pixel is targeted from the selected camera. In addition, as the project aims to use video for ball tracking, a high frame rate (higher than the regular 30 frames per second (fps) mark) is required. This is in keeping with the fact that the ball velocities are typically in the order of a few metres per second, and to track the ball at close spatial intervals, a high frame rate camera is needed. However, a brief product survey on high resolution (several megapixel), high-speed cameras showed that the camera prices increase exponentially with the resolution and frame rate, and for the requirements set earlier in this section, the cost of the camera would be around several thousand pounds.

As a compromise, it was decided to select a high-resolution, low frame-rate camera with the region of interest (ROI) option that also enables high-speed capturing of images for partial images. It was also decided to position the camera in one half of the table, divided along the lateral length of the table- connecting the middle pockets - thus not compromising on resolution but at the same time having the option of relatively higher frame rates.

A colour camera is required, because the table contains balls of different colours. Although the coloured balls have proved to be differentiated with their intensity values **[Shu 1994]**, in support of the use of monochrome camera, very uniform lighting conditions throughout the table are necessary to accomplish this task. If a colour camera is found to accurately differentiate the balls under the current lighting condition over the snooker table, there will be no need for uniform lighting, which is very costly.

Shutter opening time, which directly influences motion blur, is also taken into account when choosing a camera. Exposure time must be short enough so that the ball moves by less than one pixel during shutter opening and subsequent sensor exposure, providing a crisp image. Both CCD and CMOS cameras were considered.

Consequently, a CMOS Firewire colour camera PL-B776F from PixeLINK® was selected (shown in Figure 4.15). The image intensity of the camera can be either 8 or 10-bit depth, and it has a resolution of 2048x1536 pixels at a nominal frame rate of 12 fps. When the ROI option is used, the camera can capture images at up to 1,000 fps. A spatial resolution of 1.5 mm/pixel is possible with the camera, when its field of view covers the table area fully. Furthermore, for this spatial resolution and for a maximum ball speed of 10 m/s (this is a high-end ball velocity), an exposure time of 1.5 ms is required. The selected camera fulfils this requirement.

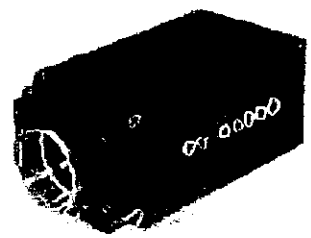


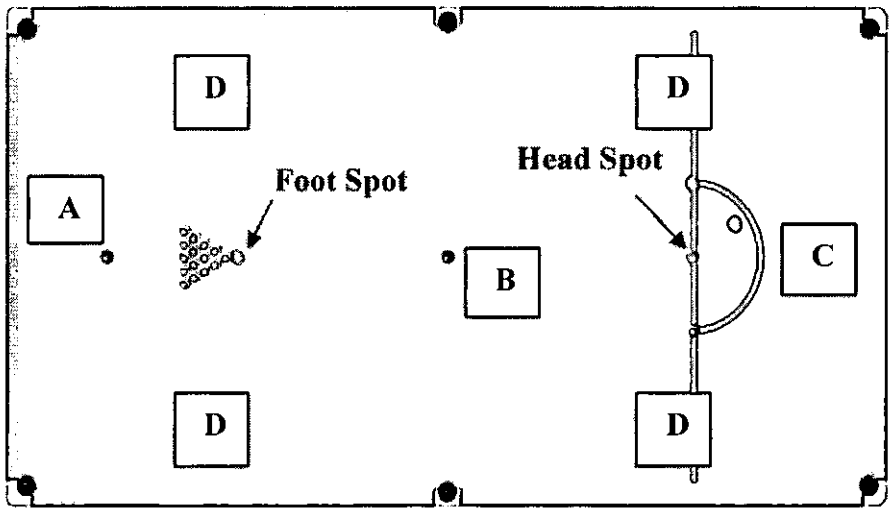
Fig. 4.15. PixeLINK® PL-B776F Firewire Camera

A ½" megapixel grade lens, H2Z0414C-MP, from computar® was chosen to be used with the camera. It has an adjustable focal length of 4-8 mm. The lens is selected based on the calculations that considered the available headspace and the table dimensions. This is a wide-angle lens according to industry norms (a lens that has an angle of view between 60° and 100°), and this lens is expected to produce high distortions in the captured images. A metal frame is attached to the joists of the mezzanine floor above the snooker table, and the camera was rigidly mounted on this frame. The camera was positioned above the table so as to face vertically down onto the table. More than half of the table is viewed by the current positioning of the camera (5 ft x 6 ft to be precise), covering the four pockets around the foot spot on that table (refer to Figure 4.16). This field of view for the camera results in a spatial resolution of 1 mm/pixel, a 33% increase compared to the camera being used to view the whole table. Moreover, as four pockets are covered by the camera field of view, the robot can be positioned to play a number of different shots, allowing it a variety of options.

4.3.1 Table illumination

For the image-processing algorithm to deliver good results, quality images should be available. The image processing software cannot compensate for the missing or inaccurate image data from the camera. Hence, the table illumination is important.

The values required for the table lighting, as per the rules in billiards, are shown in Figure 4.16 **[Marlow 1994]**. Moreover, it is suggested that the variations in the lighting between any two points on the table shall be less than 200 lumen/m^2. Two 200 W incandescent frosted bulbs located 1 m above the playing surface at the Head Spot and Foot Spot (see Figure 4.16) are recommended to achieve the above conditions **[Marlow 1994]**.



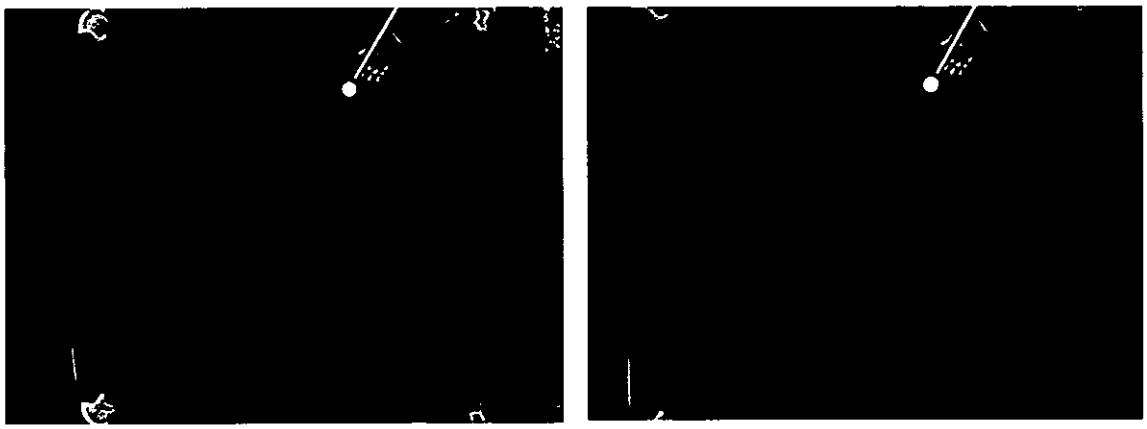
Illumination of points A, C > 190 lumen/m^2, B > 330 lumen/m^2, D > 210 lumen/m^2

Fig. 4.16. Table illumination requirements **[Marlow 1994]**.

In the present arrangement, panels of fluorescent strip lights with standard diffusers illuminate the table area, and the panels are situated asymmetrically above the table.

4.3.2 Setting up the Camera

A calibration procedure is performed to correct for barrel-type distortion that is present in wide-angle lenses (see Figure 4.17 (a) and (b)). The procedure is called intrinsic camera calibration. The camera calibration toolbox from the Computational Vision Group at Caltech is used in conjunction with MATLAB® to calibrate the camera; for a detailed description of the procedure refer to Bouguet **[2008]**.



(a) (b)

Fig. 4.17. Distorted and corrected images of 5 ft x 6 ft table area

The toolbox also incorporates an extrinsic calibration element. The extrinsic calibration procedure enables real-world measurements to be made from the values measured in terms of pixels from images.

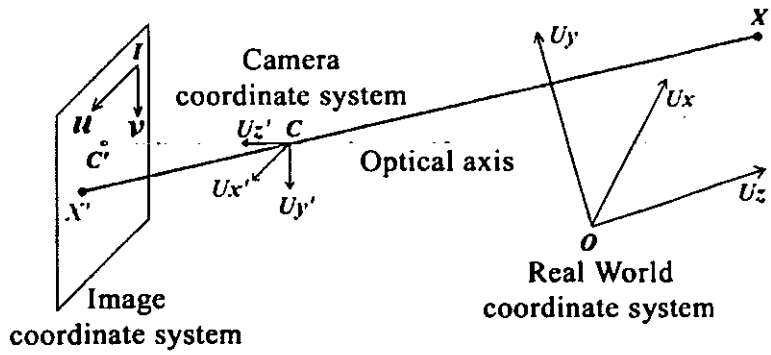


Fig. 4.18. The pinhole camera model

The extrinsic calibration procedure provides transformation (translation and rotation) matrices between the real-world coordinate system and the image plane (Figure 4.18). These matrices allow metric measurements to be made from parameters measured in the image plane. The equation for the transformation between a point in the real-world frame XX to its corresponding image point in the camera frame XX' is $[X']=[R_C]*[X]+[T_C]$, where $[R_C]$ and $[T_C]$ are the rotation and translation matrices, respectively **[Heikkila and Silven 1997]**.

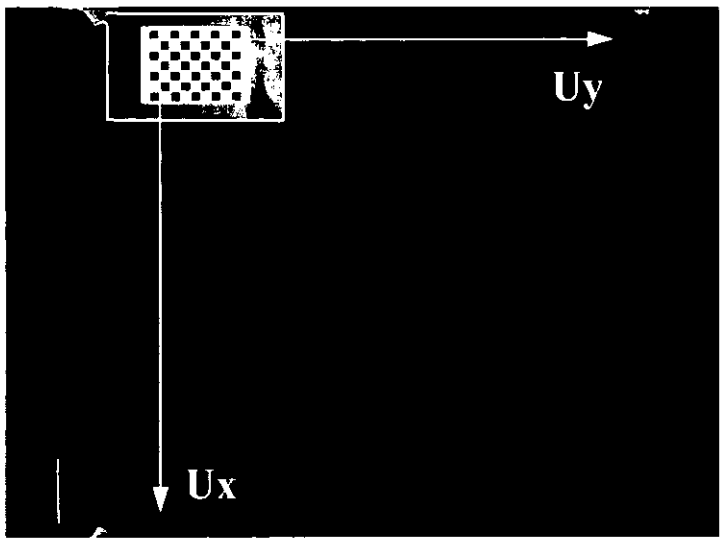


Fig. 4.19. Extrinsic calibration pattern placement (un-distorted image)

Here, the real-world coordinate system is selected such that it is fixed to the snooker table so that two of its axes lie along the two perpendicular edges of the table, and both *Ux* and *Uy* lie on the imaginary plane that is created by the ball centers, as depicted in Figure 4.19. Snooker balls have a uniform diameter of 52.4 mm; hence, the imaginary plane lies at 26.2 mm above table surface. Image blur, due to fast moving balls, is kept to a minimum by selecting the lowest possible shutter opening time available in the camera. Image sequences with high image blurs are not analyzed. This quantification is performed by counting the pixels of a moving ball and then comparing it with the number of pixels found in a stationary ball.

4.4 Cue-tip Force Sensor

Considering the nature of the geometry of the cue-cue tip interface and the force transferred from the analysis performed earlier in this chapter, a thin film (0.208 mm thick) force sensor called Flexiforce® A201-100 was chosen. The thin film sensor, firmly sandwiched between the cue tip and the cue shaft, is not expected to change the cue characteristics mentioned earlier in this chapter. The sensor can measure loads in the range of 0-4400 N **[Tekscan 2009]**, making it suitable for measuring the forces at the cue tip (see Section 4.1.3). The sensor also has a response time of less than 5 μsec making it possible to measure the impulses that have an active time in the order of milliseconds. The sensing area is circular with a diameter of 9.53 mm. The diameter of the cue at its tip is around 9 mm for the Riley® snooker cue used. The sensor is firmly glued to the wooden part of the cue and then the cue tip is glued over the sensing area of the sensor (see Figure 4.20).

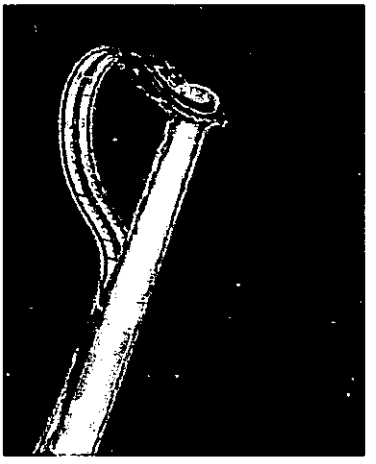


Fig. 4.20. Force sensor attached to the cue

An amplifier circuit recommended by Tekscan, Inc (Appendix I) initially conditions the data from the sensor. Data acquisition is performed using a National Instruments® (NI) 9215 sampling input module fitted with a NI USB-9162 carrier (see Figure 4.21). The sampling can be at up to 100 kS/s (100 kHz) from this data acquisition unit. Here it is operated at 10 kHz, giving 20-40 data samples during the impulse, depending on the magnitude of the impulse. The device is controlled from the main VB program as described in Section 3.2.2. The data from the device is programmed to be saved in a text file and then retrieved by the program to do the force-time integration in order to

obtain the value of the impulse. A numerical integration scheme based on the trapezoidal rule is used here. The algorithm also detects the peak force measured.

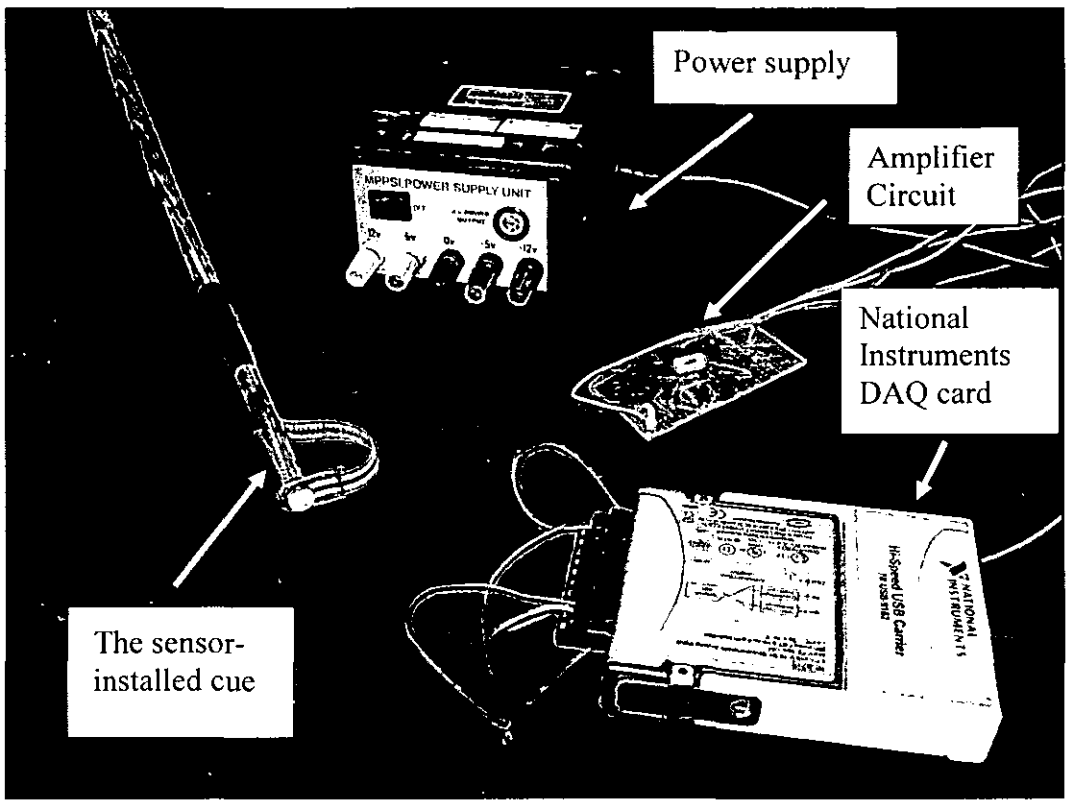


Fig.4.21. The force measurement setup

4.4.1 Force Sensor Calibration

Voltage readings out of the force sensor have to be related to the actual force by a calibration process. A mil® load transducer, of type U4000, is used to calibrate the force sensor. The load transducer has a force range of 0-1500 kgf. During the calibration, the cue is kept tightly in contact with the load cell through a little-used worn cue chalk block that ensures that the cue stays in place, preventing the lateral movement to the loading direction. An axial force is applied to the cue by holding it rigidly through a C-clamp, as shown in Figure 4.22.

When the force readings are stabilized, the force value is read out on the indicator of the load cell and the corresponding voltage out of the force sensor circuit is also measured. Using multiple readings, a calibration curve is plotted. Tekscan, Inc **[2009]**

obtain the value of the impulse. A numerical integration scheme based on the trapezoidal rule is used here. The algorithm also detects the peak force measured.

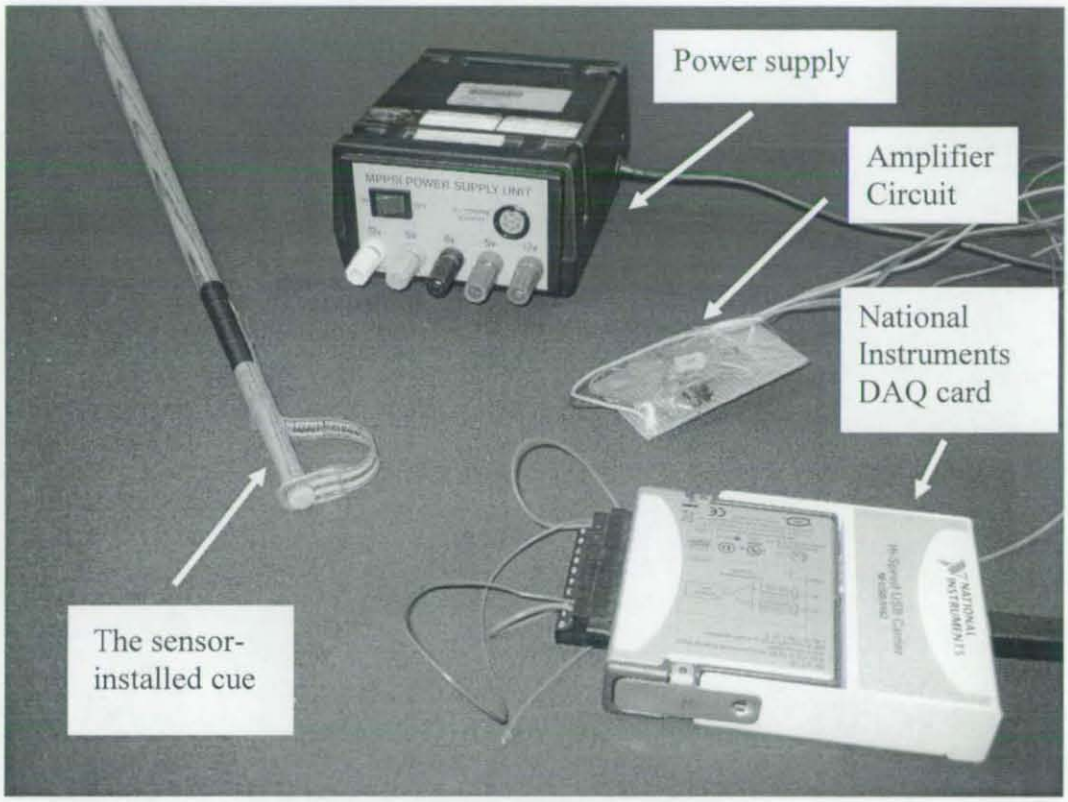


Fig.4.21. The force measurement setup

4.4.1 Force Sensor Calibration

Voltage readings out of the force sensor have to be related to the actual force by a calibration process. A mil® load transducer, of type U4000, is used to calibrate the force sensor. The load transducer has a force range of 0-1500 kgf. During the calibration, the cue is kept tightly in contact with the load cell through a little-used worn cue chalk block that ensures that the cue stays in place, preventing the lateral movement to the loading direction. An axial force is applied to the cue by holding it rigidly through a C-clamp, as shown in Figure 4.22.

When the force readings are stabilized, the force value is read out on the indicator of the load cell and the corresponding voltage out of the force sensor circuit is also measured. Using multiple readings, a calibration curve is plotted. Tekscan, Inc **[2009]**

suggests a linear variation between the output voltage of the sensor and the force measured for its Flexiforce® sensors, and this variation has also been experimentally validated by Komi *et al.* **[2007]**. The plot obtained is shown in Figure 4.23.

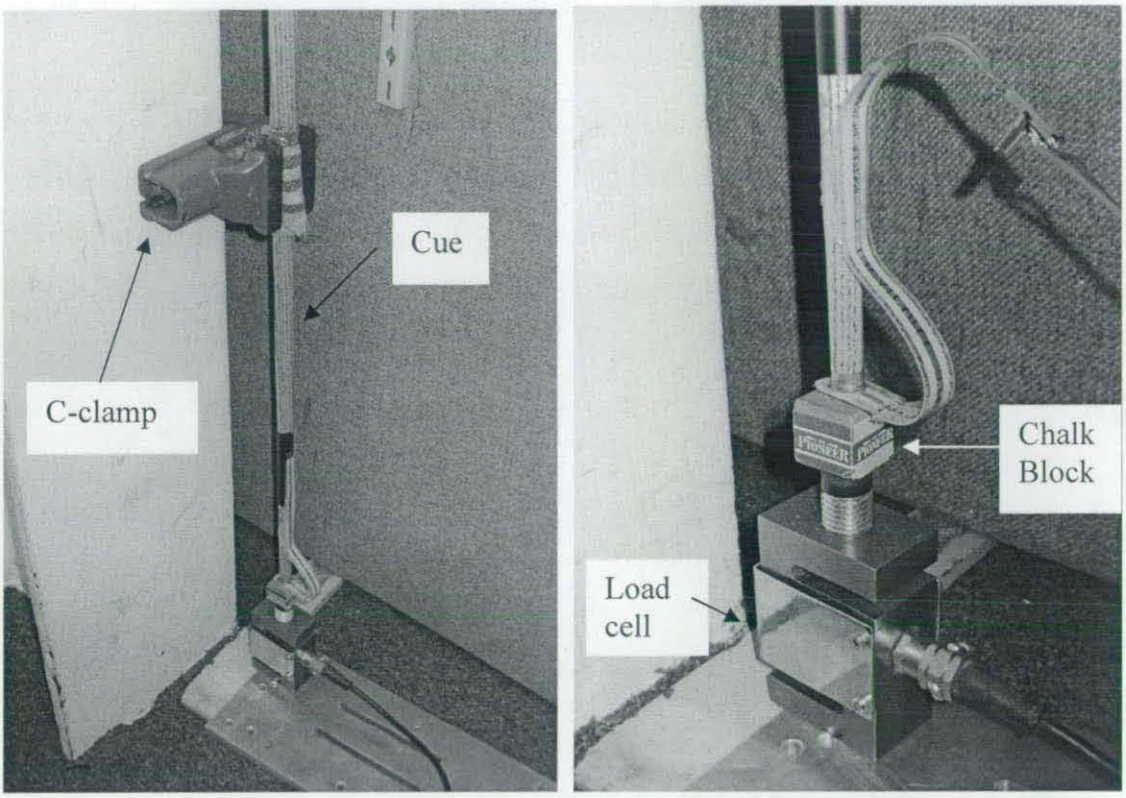


Fig. 4.22. Calibration of the force sensor using a load cell

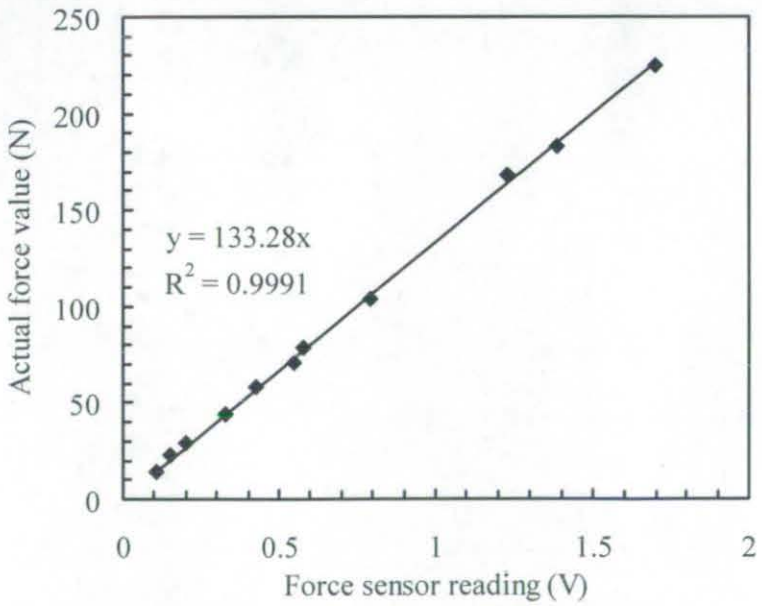


Fig. 4.23. Calibration results for the force sensor

The dynamic forces that are present at the cue-cue ball interface during collision are larger than the force range given in the calibration curve. However, only a maximum static force of around 250 N could be applied to the cue-tip, and beyond this the cue started to bend and appeared to be about to break. Dynamics forces of this magnitude do not harm the cue as they only last, at most, for few milliseconds. However, the linear variation assumption is used to estimate the value of the forces that are outside the calibration, using the gradient of 133.3 N/V obtained from Figure 4.23. Extensive tests from Komi *et al.* **[2007]** present the evidence for a linear variation for this type of force sensor.

4.5 Overall configuration of the system

The overall layout of the system is depicted in Figure 4.24. Some more photographs of the system are given in APPENDIX I.

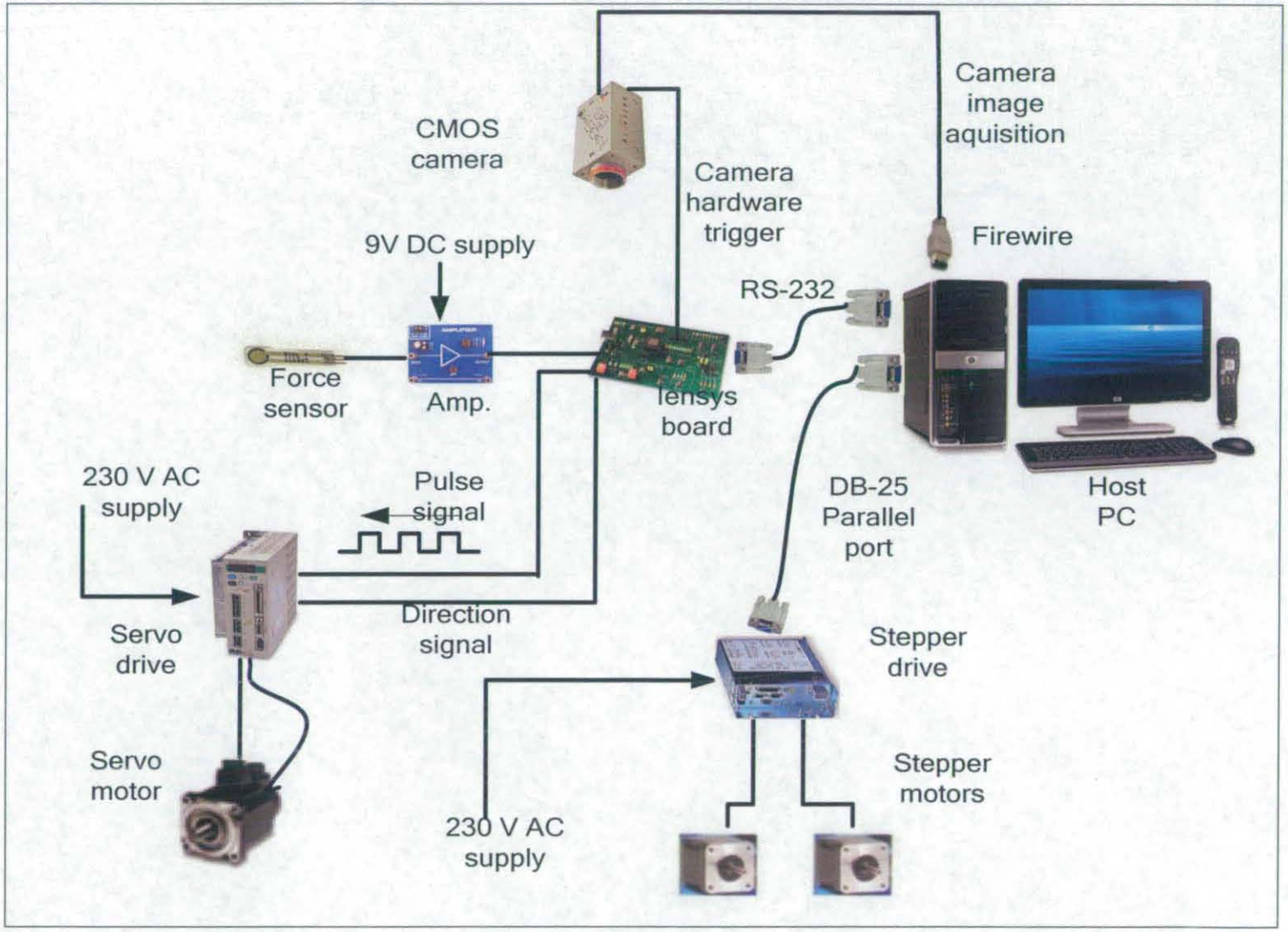


Fig. 4.24. Overall hardware configuration of the system

Summary

Here the specific details of the system development have been given. Then the design of the system is explained. The previous literature barely had any details and numerical values of the various parameters on which this design could be based. Various techniques, such as the use of relevant equations, performing some approximate reasoning on the data produced by other researchers have been used. Some 3-D drawings and the pictures of different fabricated units have also been provided here.

Chapter 5

EXPERIMENTS ON THE DYNAMICS OF SNOOKER

The literature review given in Chapter 2 shows that the values of many of the parameters related to the dynamics of snooker have not been measured comprehensively. In many cases these values are found only for the game of pool, for example, those detailed parameters given by Marlow **[1994]**. However, a system-dynamics-based approach for the determination of the ball trajectories in snooker, in order to apply appropriate robotic manipulation strategies, calls for a mathematical model for the dynamics of snooker. The mathematical model, in turn, will consist of several parameters related to the ball motion, as described in Section 2.4. This chapter describes a novel, high-speed camera, tracking-based experiment to determine the parameters involved in snooker dynamics. In addition, experimental results from the force sensor that is incorporated into the cue are also provided. A number of results of a camera-force sensor combined measurement setup are also given in this chapter. Some efforts to track the ball spin within a limited area on the table using a single circular pattern on the ball have also been reported at the end.

5.1 High-Speed Camera Based Results

High-speed tracking technologies are extensively used in sports such as football, tennis and cricket **[Pingali *et al.* 2000, Davis 2009]**. Alciatore **[2004]** has also used high speed video capture to clearly visualise the dynamics in the game of pool, where such videos are used to illustrate many principles found in pool in an inspiring way. Alciatore **[2009]** has also made use of infrared imaging to visualise the collision points. The collisions produce heat, and the associated high-temperature region is distinguished by the infrared imagery. However, Alciatore has not analysed any of the videos to extract the physical parameters involved in the dynamics of pool. Cross **[2008]**, in a very recent work on billiards, has employed a video camera to measure

the ball velocity and ball spin using an overhead camera, and used this approach to analyse squirt dynamics in a cue ball suspended as a pendulum bob.

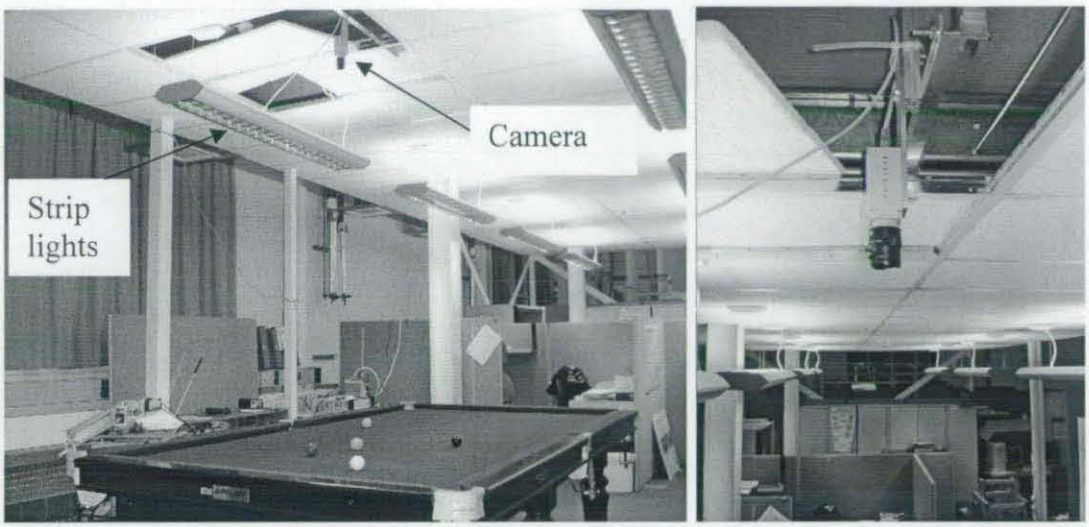


Fig. 5.1. The ceiling-mounted machine-vision camera in the mechatronics lab

The research reported in this thesis employs a machine-vision camera, as explained in Section 4.3. The camera is mounted overhead on the ceiling, right above the snooker table, looking vertically downwards (see Figure 5.1). To independently verify that the measurements made by the camera are accurate, prior to the actual measurements, some distance measurements were also made with a metre rule. For this purpose, two rectangular blocks, having a height of half the ball diameter, with circular white patterns on their top surfaces were placed at two different locations on the table. Circular patterns of diameter 52.4 mm (i.e. the ball diameter) were used so that the camera and the image-processing algorithm would treat them as balls. The distance between the centres of the circular patterns was obtained both from the camera and by using the metre rule. This method was used because it was very cumbersome to physically measure the centre distance between two random snooker balls due to the balls changing position even with the slightest touch. This procedure was repeated for several random positions of the blocks covering the 5 ft x 6 ft area of the table that is imaged by the camera. The differences in measurements by the two methods were found to be at most 2 mm, and the metre rule measurements are subject to ±0.5 mm error. Hence the ball centroid measurement is performed to an accuracy of ±0.5 mm.

The pre-processing of the video acquired by the camera was performed using MATLAB®. The flowchart of the algorithm is shown in Figure 5.2. The camera calibration procedure is described in Section 4.3.2.

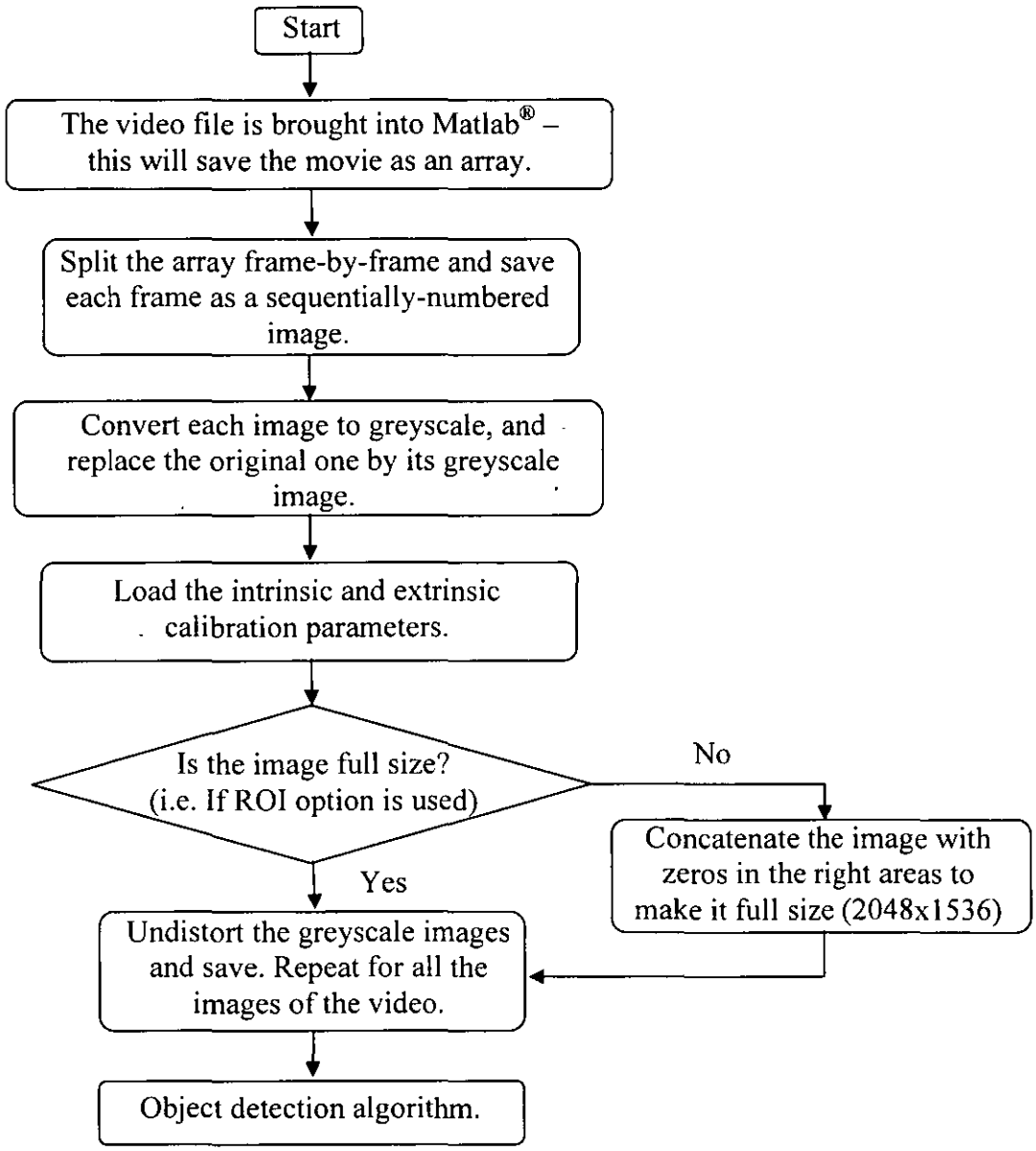


Fig. 5.2. Algorithm for the pre-processing of an image

Two standard functions *bwlabel* and *regionprops* within the MATLAB® Image Processing Toolbox were used to extract the ball from the image, and then to extract the ball centre in pixel values. Finally, the real-world coordinates of the ball centroid are obtained using the transformation matrices, *Rc* and *Tc*, as obtained from the extrinsic calibration procedure (refer to Section 4.3.2). The time-stamping of the

spatial positions of the ball centres, based on the camera frame rate of capture, enables the calculation of the velocities and accelerations of the balls.

The combined repeatability of the camera and the image-processing algorithm is found to be 0.1 pixels, on average, when it comes to detecting the centre of the ball. Since the camera images the table area to a spatial resolution of 1mm/pixel, the spatial repeatability of the imaging system is 0.1mm, as far as the ball centre detection is concerned. The inconsistency in the ball centre detection is mainly due to noise problems associated with the image sensor and due the inconsistencies involved in the table illumination.

Now, the following equation can be written to evaluate the ball speed,

Ball speed (in metres/second) = Distance travelled between successive image frames (in metres) * Camera fps (5.1)

Usually, the first image frame in which the ball has started moving commences the image sequence that is used for the tracking. When equation (5.1) is used for high frame-rate capture, due to the camera repeatability in the order of one tenth of a millimetre, changes in the tracked positions are magnified and "error speeds" are produced. The obtained "error speed" values were up to a maximum of 0.05 m/s at 200 fps camera capture rate. However, this value is relatively small when compared to the normal ball speed values that are encountered in snooker.

Figure 5.3 shows the tracked cue ball positions of an extremely high-speed shot superimposed on the image that was captured at the start of the tracking, also shown is the initial cue ball location. White markers denote the successive centroids of the cue ball.

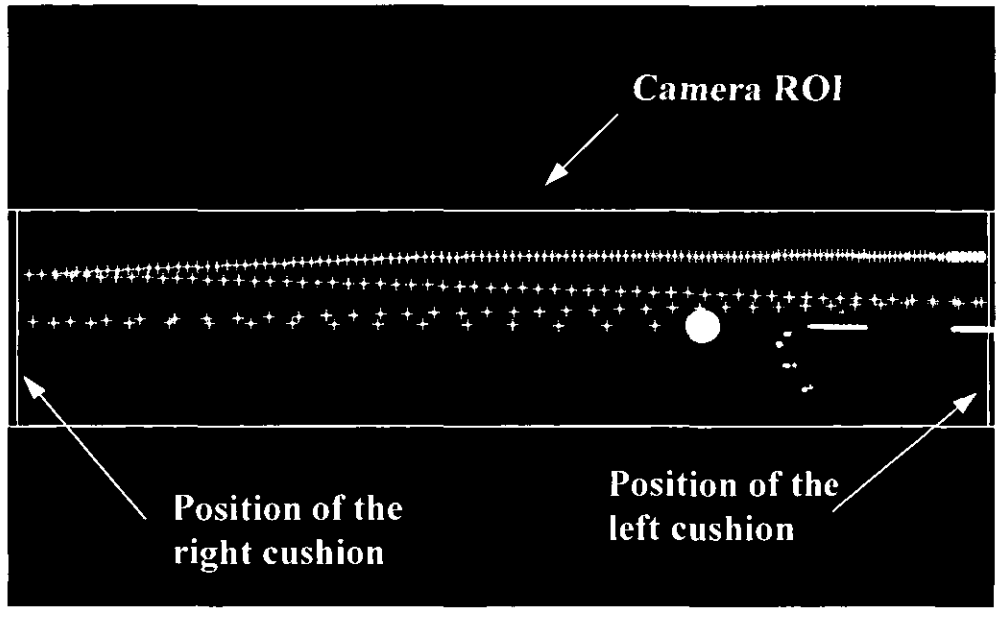


Fig. 5.3. Tracking the cue ball (4 consecutive impacts within two parallel cushions are shown)

5.1.1 Friction Coefficients

5.1.1.1 Rolling Friction

Figure 5.4 shows the variation of the ball velocity with respect to time (the complete motion profile until the ball comes to rest is not shown here). As seen in Figure 5.4, once the impulse is delivered to the ball, the ball velocity decreases rapidly, during the sliding phase as described later, and then the ball starts to roll. The velocity gradient during the rolling phase gives the value of the deceleration due to rolling friction.

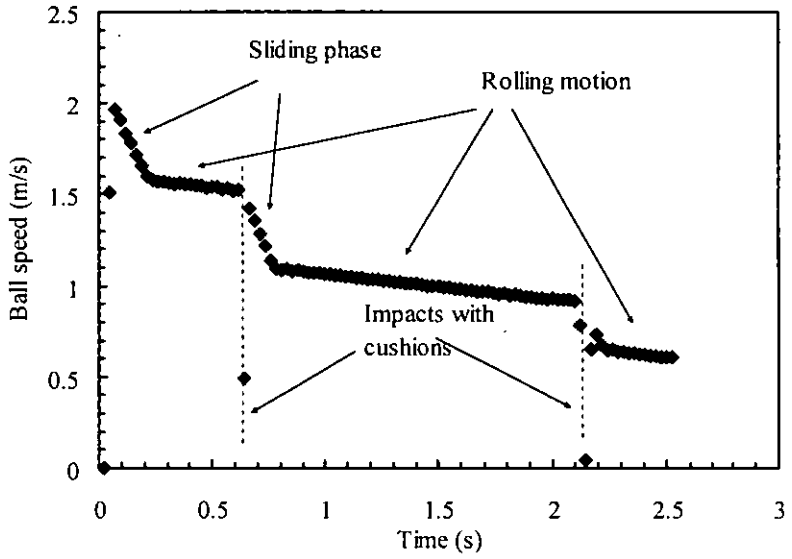


Fig. 5.4. Speed-time plot for the ball showing different phenomena involved from the video captured at 42 fps

Different shots were tracked and the deceleration during the rolling phase was found to be very consistent between 0.124-0.126 m/s^2. The rolling friction as a non-dimensional number (in relation to the gravitational acceleration g) is evaluated as 0.0127-0.0129. Marlow **[1994]** suggests a range of 0.011 to 0.024 for the game of pocket billiards (widely known as pool), and suggests a mean value of 0.016. Here it must be noted that the physical properties of the ball and table in pool and snooker are different. However, there is no clear reason for this excessive variation (more than 100% of the lower value) obtained in pool using Marlow's measurements. The only plausible explanation is that the metre stick and stopwatch measurement method that was used by Marlow is very limited and excessively prone to errors due to human judgments. Although Williams **[2002]** claims that the nap of the table felt affects the ball motion, depending on whether its motion is toward the top cushion or away from it, no evidence was found in the results to substantiate this claim.

5.1.1.2 Sliding Friction

The ball speed-time plot given in Figure 5.4 shows that the sliding friction is much larger than the rolling friction. In addition, the sliding phase is also shown to disappear within a very short period of time, but it diminishes the ball velocity considerably (Figure 5.4). Another interesting observation from this plot is that, once

the ball has entered the rolling phase, when it collides with the cushion (wall/rail) it starts to slide again (note the speed gradients immediately after the cushion impacts), because the cushion impact breaches the $V=R\omega$ condition that is attained by the ball during rolling. Therefore, the ball starts to slide again. Once the condition $V=R\omega$ is reached, the ball again starts rolling.

From the analysis of the speed of the tracked ball, the sliding friction coefficient was found to be in the range of 1.75-2.40 m/s^2 (0.178-0.245 non-dimensionally). These values were obtained for the ball motion along different, random directions on the table. Hence, the average non-dimensional value of 0.21 will be used from here onwards for the sliding coefficient of friction. Marlow **[1994]** suggests a non-dimensional value of 0.2 for pool; Marlow calculated this value from certain theoretical derivations, and, in the process, also made use of the rolling coefficient value of 0.016, as seen in the previous section. An independent measurement was not performed, because only a metre rule and a stopwatch were used by Marlow. Witters and Duymelinck **[1986]** use a technique similar to the camera-based tracking reported in this research in that stroboscopic illumination is used to photograph a decelerating pool ball. They say that the sliding friction coefficient varies between 0.14 and 0.21, and that when the ball velocity increases from zero, the friction coefficient asymptotically approaches 0.21 from a value of 0.14. However, no such variation could be obtained from the present experiments.

Notably, the sliding friction is 15-20 times larger than the rolling friction. Also, during the sliding phase, some rolling action will simultaneously take place, as the 'ball sinking' effect is ever-prevalent at the ball-table interface. However, due to its comparatively small magnitude (approx. 1/17), it is usually neglected and the motion is treated as pure sliding.

5.1.2 Ball-Cushion Impact

To visualise and analyse the impulse dynamics between the ball and the cushion, a series of high-speed image capturing experiments, using in excess of 100fps with very small ROIs, was performed. The cue ball was tracked in all these experiments.

Generally, the ball dynamics after a cushion impulse depend on several factors. Some of the factors are the incident speed at which it collides with the cushion surface, the incidence angle with respect to the cushion surface, the amount of spin the ball has, the physical characteristics of the ball, the cushion and the parameters involved in the interaction between them, such as the coefficient of restitution or the surface friction.

Spin on the ball changes the impact characteristics drastically. The ball spin is difficult to quantify with the present experimental methodology where only the ball centroid is tracked by the camera. Sidespin especially changes the post-impulse cue ball path significantly as explained earlier **[Walker 1983, Alciatore 2004]**. The ball-cushion interaction is a case of multiple impacts, both normal and tangential, the latter due to friction, simultaneously acting on the ball in the 3-dimensional space, one normal to the cushion surface and the other two perpendicular frictional impacts from the cushion wall. Theoretical derivations for the dynamics of general impact are not found in the literature. Section 6.2 will try to address this problem.

For this reason, it was decided to conduct experiments on shots without considerable sidespin. Every time a shot was made, care was taken such that it was directed perpendicular to the cushions as much as possible. Whenever the cue ball is played perpendicular to the cushions, if it does not have any sidespin, it bounces back along the same line along which it approached the cushion. This criterion was used to ensure that the analysed shots did not impart a considerable sidespin on the cue ball. Figure 5.5 (a) shows a perpendicular shot with no sidespin, and Figure 5.5 (b) shows a perpendicular incoming shot that apparently has some side spin, which results in the ball rebounding to the right side. Thus, for the rebound analysis, only the shot given in Figure 5.5 (a) will be used and the shot shown in Figure 5.5 (b) will be discarded. This procedure ensures that there is only one unknown in the form of top/back spin. The top/back spin of the ball is estimated by the method outlined below.

Assuming that the ball had gone into pure rolling mode before the collision with the cushion (this can be determined by the speed-time plot gradients as explained in the previous section), the topspin of the ball can be calculated by the formula, $\omega = \frac{V}{R}$, where R is the ball radius. Thus the incident ball speed, V, is the only independent

variable involved and the velocity drop during the impact can, theoretically, be correlated to V.

(a) (b)

Fig. 5.5. Bounce of the cue ball from the cushion, the ball location depicts its position as it approached the rail (at 120 fps frame rate)

Figure 5.6 shows the speed plot obtained for a high-speed video captured at 150fps. The speed plot itself was used to determine if the ball was rolling just before it collided with the cushion, by evaluating the value of the gradient of the speed-time plot immediately before the collision. Results that were obtained for 31 such shots into the cushions, satisfying the conditions imposed above, are given in Figure 5.7.

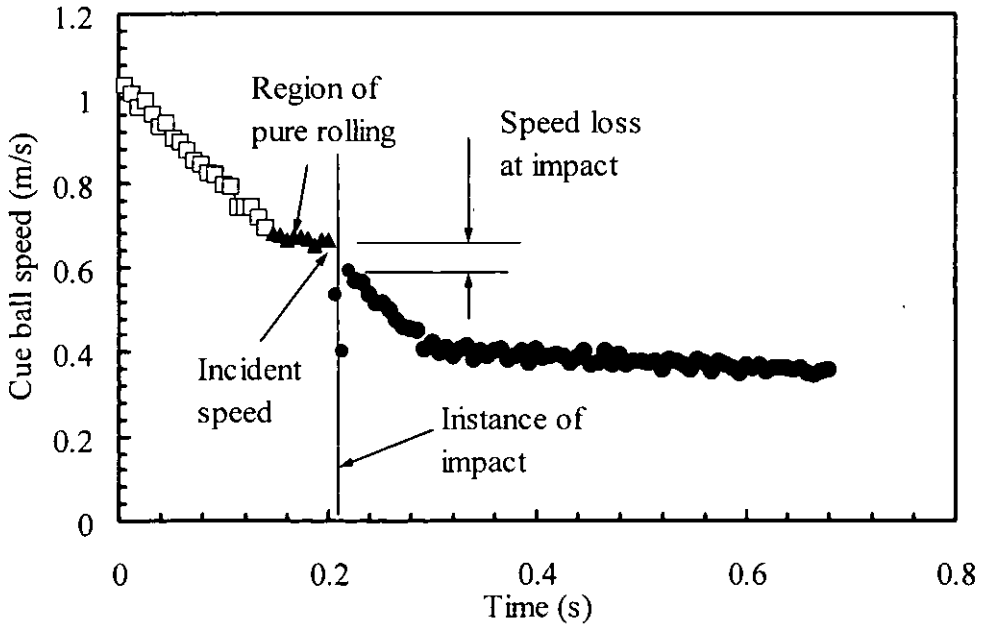


Fig. 5.6. Results based on tracking for a ball-cushion impulse (at 150 fps)

From Figure 5.7 it can be seen that the relationship is almost linear for the incident velocity in the range of 0.28 – 3.5m/s (this is the typical range of ball velocities in the game). A best-fit straight line gives a coefficient of restitution of 0.818 for this velocity range. But the results have a better fit with a 2nd order polynomial of $y = -0.0877x^2+1.131x-0.0953$ in the 1st quadrant, where x is the incident velocity and y is the rebound velocity. Here, it must be kept in mind that these results are not valid for a general ball-cushion impulse but applicable only under the conditions of no sidespin and pure rolling motion prior to the impulse.

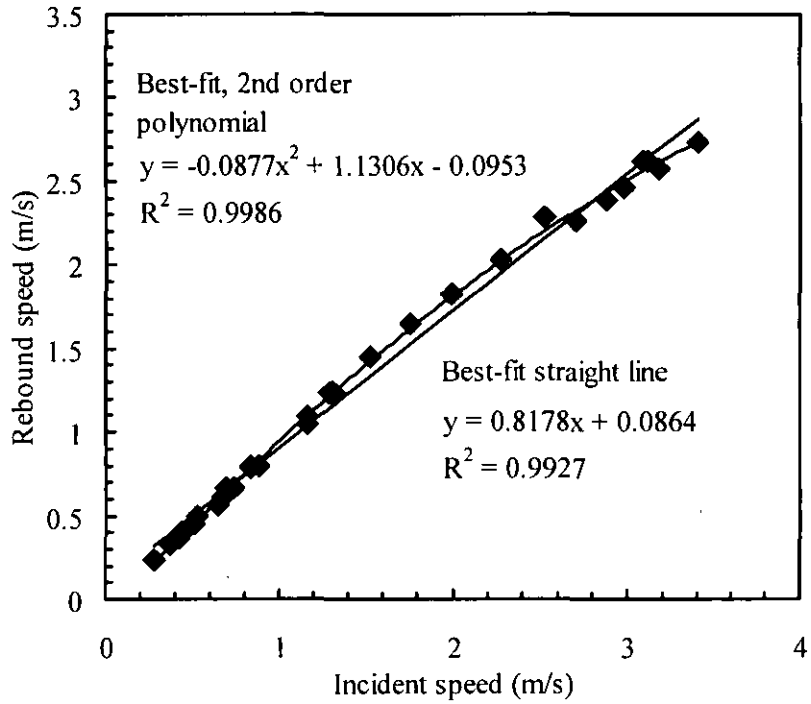


Fig. 5.7. Variation of the rebound velocity against the incidence velocity

Marlow **[1994]** reports that the coefficient of restitution is 0.55 for cushions in pool. Here again the results are reported without much detail about the experimental procedure. However, Marlow compares the results with the values suggested by Coriolis and concludes that they agree very closely **[Marlow 1994]**. The cushion height for snooker is 36 mm, with the ball radius being 26.2 mm, and this closely corresponds to the height value of 1.4 times ball radius found in pool. Thus the cushion and ball geometry is almost identical in pool and snooker. It is possible that, in order to calculate the coefficient of restitution, Marlow considered the rebound ball velocity at the end of the sliding phase rather than the correct one immediately after

the impulse, possibly due to the limitations with the experimental methods. In this way, from Figure 5.6, the coefficient of restitution for the shot will be around 0.63, but this value is without any real physical meaning.

This stringent experimental procedure exercising a tight control over impact conditions will enable the estimation of some more parameters in the cushion-ball impact, as seen in Section 6.2. This, when combined with a numerical model for the impulse dynamics, will permit the calculation of the ball trajectories off the bounce as seen in Section 6.2.

5.1.3 Impact between Balls

Wallace and Schroeder **[1988]** in their analysis of collisions between billiard balls assume that the balls are perfectly elastic (e=1). However, they neglected the effect of the friction between the billiard balls. By considering the friction effect from the table, they derive that, for the object ball (see Figure 5.8),

$$V_O = \frac{5}{7} V \cos\theta \text{ and } \theta_O = \theta \tag{5.2}$$

and for the cue ball,

$$V_C = \frac{5}{7} V \sqrt{\frac{9}{5}\sin^2\theta + \frac{4}{25}} \text{ and } \theta_C = tan^{-1}\left[\frac{\sin\theta.\cos\theta}{(sin^2\theta + \frac{2}{5})}\right] \tag{5.3}$$

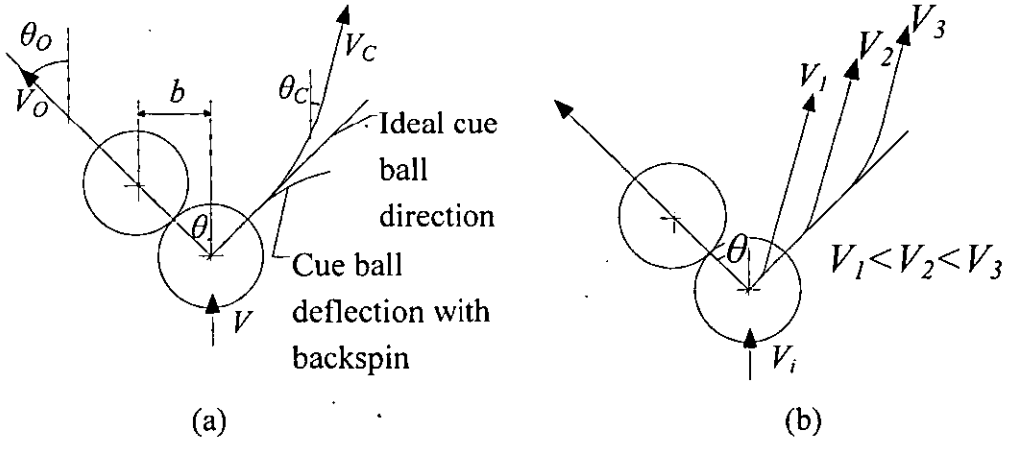


Fig. 5.8. Ball collision with the effect of table friction

Wallace and Schroeder also define β as the fractional impact parameter, and $\beta=b/D$, where D is the ball diameter and b is the separation of the ball centres in the direction perpendicular to the incident ball velocity V, as defined in Figure 5.8(a). Also note that $\beta=\sin\theta$.

Here both the predicted velocity and its direction are different to those from the analysis that does not account for the friction effects from the table. Physically speaking, the curving (see Figure 5.8) occurs because, the collision only reduces linear velocity and not the angular velocity, and the result is that the cue ball attains a sliding condition with excess topspin (called 'overspinning'). The ball initially moves in the ideal direction as shown in Figure 5.8(a), but the spin causes it to accelerate and curve forward until a rolling condition is reached **[Onoda 1989]**. Figure 5.8(b) shows the effect of the pre-hit cue ball velocity on the deflected path of the cue ball. If the ball has some backspin (draw) prior to collision, the cue ball moves to the opposite side of the ideal line.

Some tests were performed to check the effectiveness of the predictions made by equations (5.2) and (5.3) and these are reported in a research paper co-authored by the present author. The paper is provided in Appendix III. However, equations (5.2) and (5.3) do not include effects such as the friction between the balls. Moreover, the equations also ignore the effects of ballspin on the collision. In addition, the two equations do not predict the amount of ballspin after the collision. Nevertheless, ballspin is vital for predicting the subsequent motion of the balls after their collision. An all-inclusive theory of ball collisions will be presented in Section 6.1.

5.1.3.3 Head-on Collisions

The cue ball and the object ball were made to collide head on (b=0 in Figure 5.8) to measure the approaching speed of the cue ball and the initial speed of the object ball after the collision. A red ball was used as the object ball. By separating the R component of the RGB image and then by processing the R component of the image, the successive centroids of the red ball were extracted (using a similar procedure to that for the cue ball). Then the speed-time plots for both the balls were plotted together. By observing the gradient of the speed-time plot for the cue ball prior to the collision of the balls, only the collisions where the cue ball was rolling before

colliding were considered. A similar procedure was followed for the ball-cushion collision as well, in Section 5.1.2. The tracking plots were used to establish that the cue ball had not had any sidespin before colliding with the object ball. The deciding criterion being that the pre- and post-collision directions of the cue ball and the direction of movement of the object ball must all be in a straight line. During the experiments, it had not been possible to obtain this condition, perfectly, all the time. The shots that closely satisfied this condition were the only ones considered for analysis (see Figure 5.9). The cue ball velocity prior to the impact was measured (the impact time was determined using the speed/time plot as performed for the ball-cushion collision), together with the object ball velocity immediately after the impulse.

Figure 5.9, shows the variation of the object ball velocity immediately after the collision against the cue ball speed immediately prior to the impact. A nominal restitution coefficient of 0.95 can be defined for the head-on impacts, under no-sidespin and for the rolling condition of the cue ball, as explained above. Later, in Section 6.1, the effects of friction on the nominal coefficient of restitution value obtained above will be given. Figure 5.9 will also be used in Section 6.1 to obtain the values of some additional parameters.

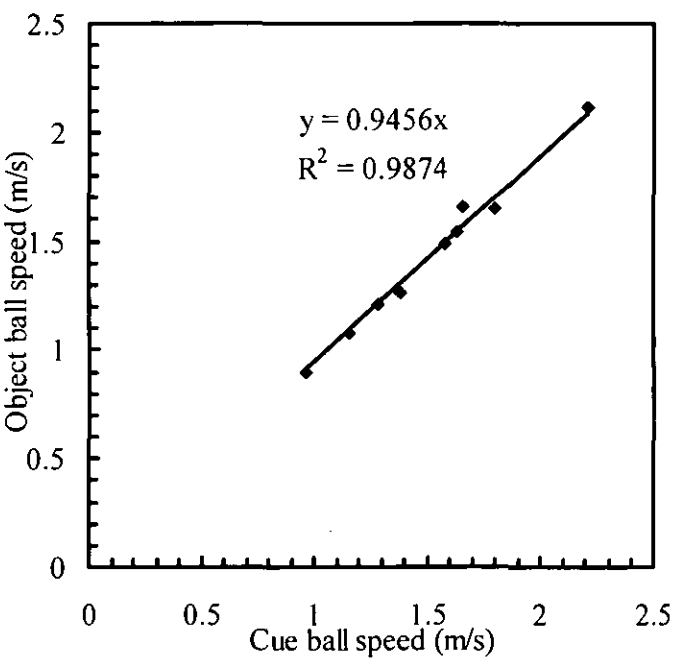


Fig. 5.9. Object ball speed against cue ball speed for stun shots

5.1.4 Cue Tracking

In order to establish the typical cue velocities for human cueing, the cue was tracked using the overhead camera. During the design stage, the cue velocity was measured approximately from the videos available from Alciatore **[2008]**. The measurement of the cue velocity will also enable the estimation of the cue velocities produced by the cue launcher. Using the same principles that were used earlier to do the metric measurements on the ball motion, and whenever the cue is kept at the level of the centre of the ball - as horizontally as possible - the camera-based measurements can be used to measure the displacement of the cue as well.

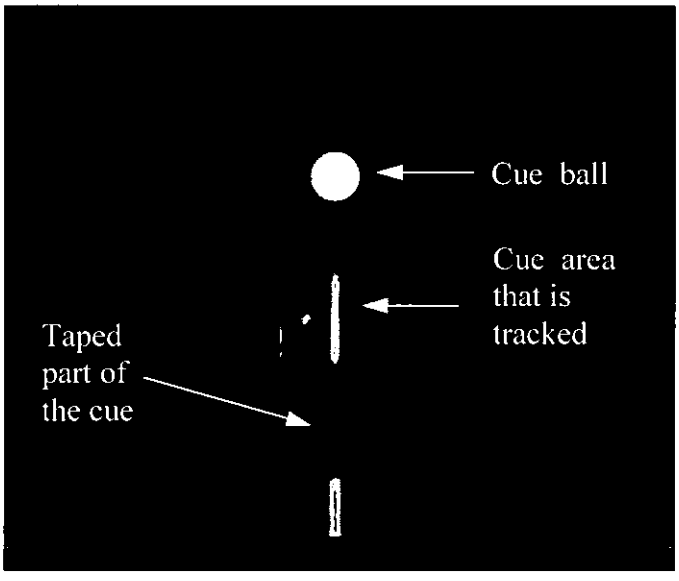


Fig. 5.10. Region of the cue that is tracked

A Riley® standard snooker cue, that that has been in occasional use for a year was used for these tests. As it was not possible to capture the cue in its full shape always because of its large length, the cue was taped using black sticky-tape, in order to capture a consistent region of it. Thus, a very small bright region of length 85mm was exposed close to the tip of the cue (see Figure 5.10). An image-processing algorithm was suitably created to detect and capture the centroid of the tracked cue region. This algorithm was very similar to that created for the ball centre detection. A typical speed-time plot obtained for the cue and the cue ball motion is given in Figure 5.11.

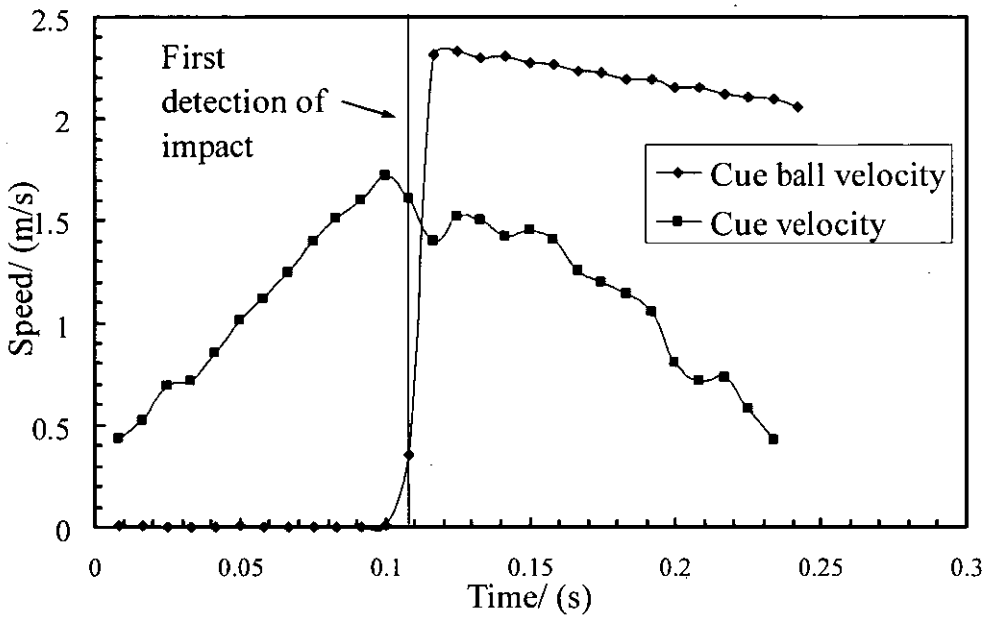


Fig. 5.11. A combined speed-time plot for the cue-cue ball impact (120fps video resolution)

In addition to the overhead camera, a PROSILICA® EC650C, Firewire, colour camera was used during these tests to horizontally view the cue–cue ball impact as shown in Figure 5.12. This camera was used to locate the point along the vertical at which the cue hit was made. Hence, the images from the horizontal camera will give an indication whether the shot imparted topspin/stun/bottomspin to the ball. The overhead camera imagery, as shown in Figure 5.12, was employed to determine the horizontal point of impact on the cue ball. Determination of the horizontal point of impact establishes if sidespin was imparted to the cue ball or not.

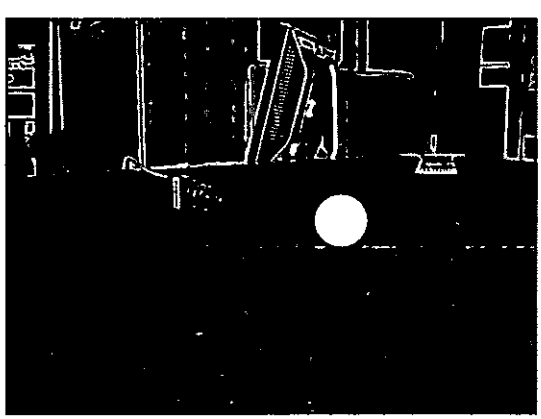


Fig. 5.12. A horizontal viewing camera to locate the vertical point of cue impact on the ball

The variation between the cue ball speed and the cue speed is usually linear, as shown in Figure 5.13 for a large number of results. The overall coefficient of velocity transfer is around 1.4. Here, it must be noted that with human cueing it is not possible to keep the cue inclination with the horizontal constant every time a shot is played. However, the cue inclination with the horizontal was kept very shallow, i.e. at the lowest angle possible. Figure 5.13 also differentiates between the stun shots and the shots with considerable spin. For stun shots the velocity transfer coefficient (1.45) is marginally higher than the shots with off-centre hits (1.35). This phenomenon is explainable, because for a spin shot only a component of the cue speed is converted into the linear speed of the ball along the direction of motion of the cue, as also explained by Cross **[2008]** in relation to the force transfer during the impact (see Figure 2.11). Moreover, Figure 5.13 also reveals an important dynamic characteristic regarding cueing; the cue ball speed is mainly determined by the cue speed. In addition, this characteristic also suggests that force measurement at the collision point may not give any additional information for the question of robotic manipulation.

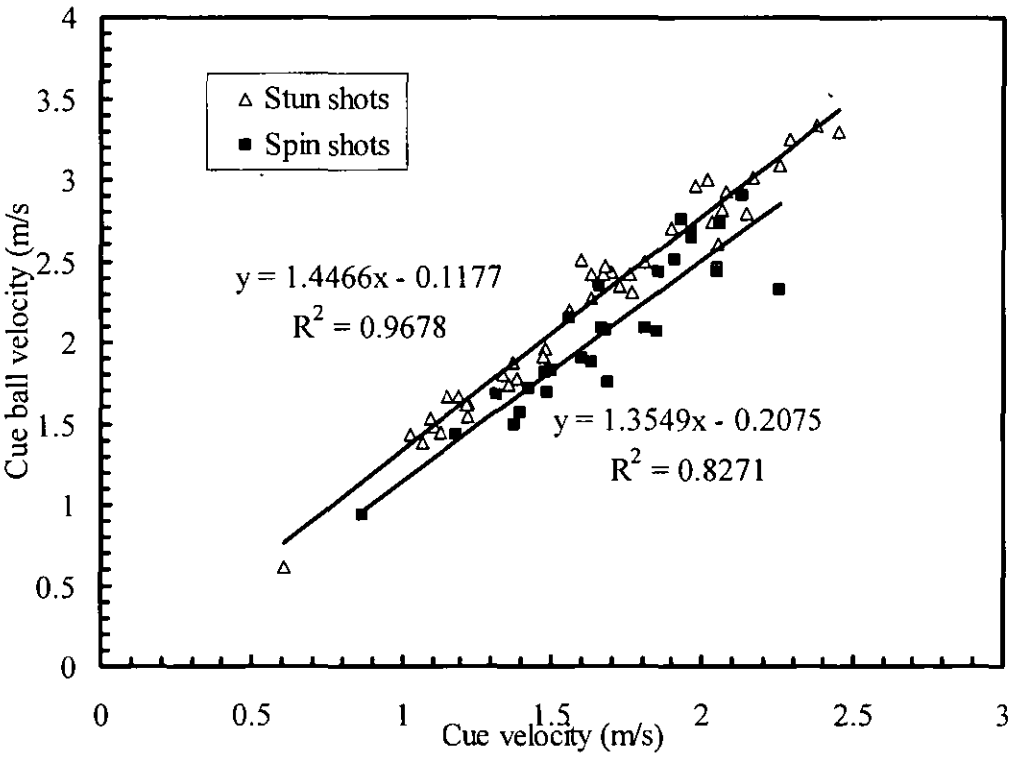


Fig. 5.13. Initial cue ball speed against cue speed at the time of impact for human shots

5.1.4.1 The Robot's Cueing Performance

In order to compare the cueing performance of the robot with that of human cueing, a number of high-speed video tests measuring the cue and cue ball speed were carried out. The robot's cue inclination angle was kept at around 2° with the horizontal. The cue angle during the human cueing was also kept close to this value. Only stun shots were played. The results of the tests are given in Figure 5.14. The figure shows that the robot is performing on a par with human shots. This performance is obtained in spite of the differences in inertia between the two; it is calculated that the inertia value of the moving components of the motor-gearbox-pinion-rack combination is 5 times that of the estimate of 1.13 kg for humans (see Section 4.1.7). Here, it must also be noted that the torque control option of the motor was not employed. Hence, it can be speculated that cueing dynamics solely depend both on the cue and its drive speed and are independent of the inertial properties of the cue-driving mechanism and the driving force of the cue. Thus, for a given cue only the cue speed and the hitting spot on the cue ball will determine the cue ball speed. This argument also eliminates the idea that the force sensor can provide some additional information about the dynamics of cueing.

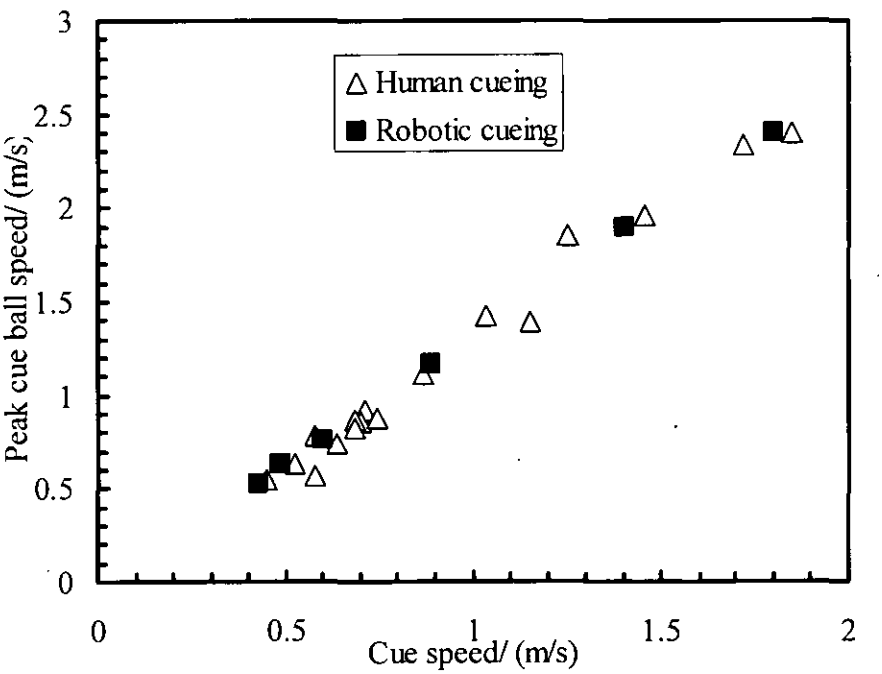


Fig. 5.14. Cue speed cue ball speed variation for human and robot cueing

5.2 Force Sensor-Camera Combined Experiments

As explained in Section 4.4, the force sensor readings during the cue-cue ball impact were recorded into the main VB program via a National Instruments® data acquisition module. After a number of trials, a data acquisition rate of 10 kHz was deemed sufficient to describe the force variation adequately. The peak force and the value of the impulse were identified by the VB algorithm. Using the calibration plot given in Figure 4.23, the measured voltage output was converted to the corresponding value of force (see Figure 5.15 for a typical plot). The area under the plot in Figure 5.15 provides the magnitude of the impulse. This value is directly related to the momentum that is transferred to the ball; hence it is an important dynamic parameter. The order of the forces obtained here is of the same order as the ones considered in Section 4.1.3 for the design of the cueing mechanism. However, the forces obtained with the real cue are around 1.5~2 times less than the values measured with the impulse hammer.

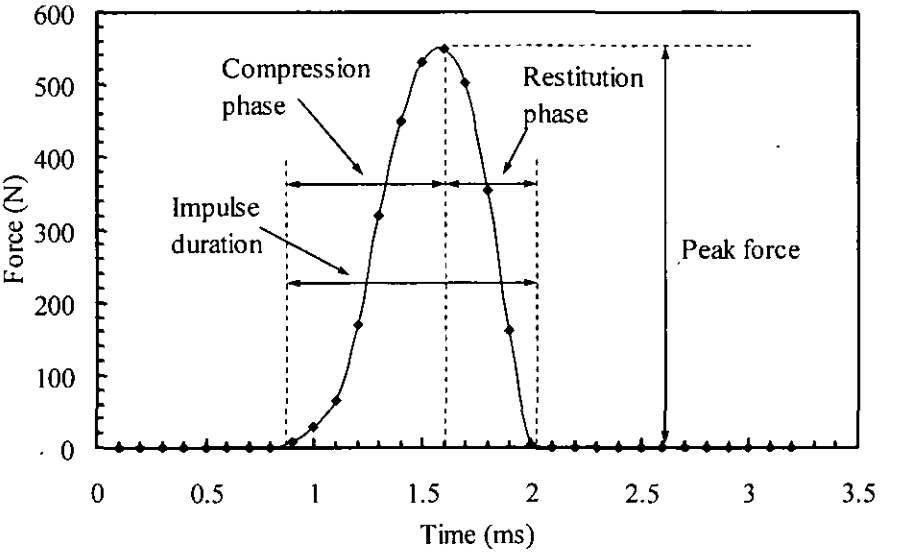


Fig. 5.15. A typical force sensor output (3.2m/s cue ball velocity) based on the calibration results given in Figure 4.23

The high-speed camera and the force sensor were also used in conjunction with each other to obtain some of the dynamics of the cue-cue ball impact.

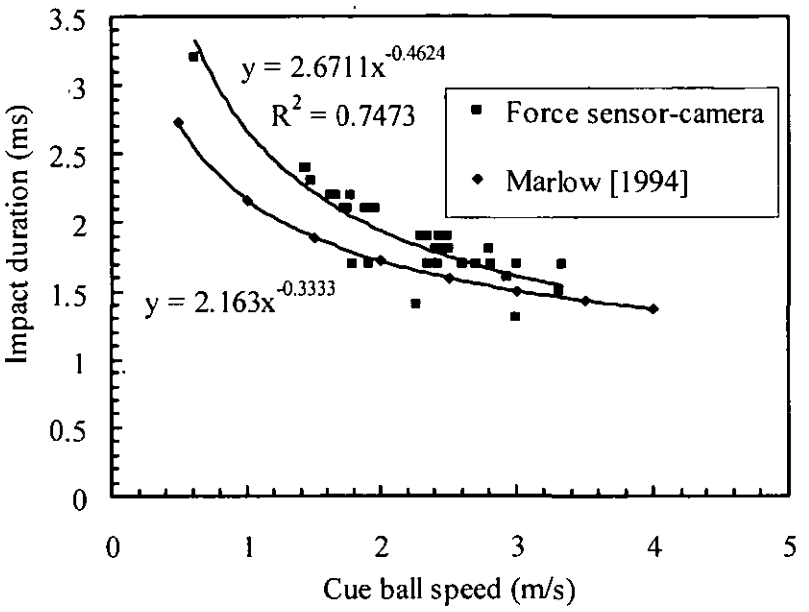


Fig. 5.16. Impact duration versus cue ball velocity

Figure 5.16 shows the impact durations recorded for various cue ball speeds (the best fit power curve is 2.67x (cue ball speed)$^{-0.4624}$). Slower shots tend to have larger durations of impact as also suggested by Marlow **[1994]** whose experimental results, in the form of a power curve of 2.16x (cue ball speed)$^{-(1/3)}$ are also plotted in Figure 5.16. Marlow admits the possibility of considerable errors in the speed variable in the experiments **[Marlow 1994]**. These errors may be the reason for the differences found.

5.3 Measurement of the Table's Resistance to the Sidespin of the Ball

The measurement of ball spin will make it possible to experiment on phenomena such as rolling, sliding and impacts that are greatly affected by friction. For example, in Sections 5.1.2 and 5.1.3.3 due to the inability to quantify sidespin, experiments were carried out ensuring that sidespin was zero, using tracking plots. However, the scope of this project does not include extensive spin measurements. Spin tracking usually involves placing a number of markers on the surface of a ball **[Griffiths *et al.* 2005, Neilson *et al.* 2004]**. To capture, differentiate and determine the spatial locations of these patterns a considerable number of image pixels has to be dedicated to the area of the ball on the image. The overhead camera that is positioned to image half the table area scarcely satisfies this requirement; only 54x54 pixels cover the ball area under the current spatial resolution. Moreover, consistent spin measurements necessitate a uniform lighting over the table area, which is not the case here. In addition, the pattern should not affect the surface properties of the ball; otherwise, the measurements will not represent the actual dynamics that take place during normal play.

As seen at the start of this chapter, a system-dynamics-based solution to robotic manipulation in snooker needs several parameters to be identified. The instantaneous value of sidespin of the ball is such an important quantity, as this is known to affect the collisions between balls and that between a ball and the cushion. It has also been noted in Chapter 2 that the sidespin of the ball is an independent entity and is dissociated from the linear velocity and topspin of the ball, leading to the assumption of 'decoupled' motion. In addition, it was also observed that there could be deficiencies in such an assumption, as the ball is known to 'sink' into the table cloth, highlighting the need for an elaborate theory in this regard. However, in the absence of such a sophisticated theory, the only available option is to use the simple current model of decoupled motion. The assumption that linear motion does not affect the rotation about the vertical, allows quantifying the instantaneous value of sidespin by a single parameter value. This parameter is the angular deceleration of the ball due to the resistance offered by the table surface to the ball rotation about the vertical.

In order to track the ball spin, a circular piece of black-coloured adhesive tape of diameter 13mm was placed on the ball surface (see Figure 5.17).

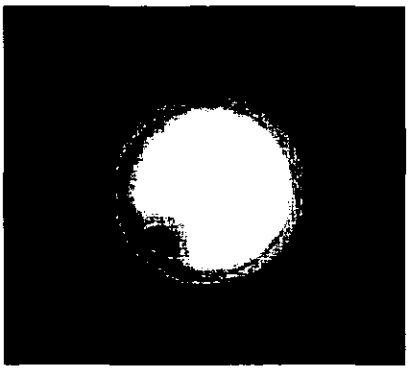


Fig. 5.17. Circular pattern on the ball surface: an image from the overhead camera

Unlike in the tracking of the centroid of the ball, when it comes to the tracking of the circular pattern on the ball the orientation of the camera has to be considered.

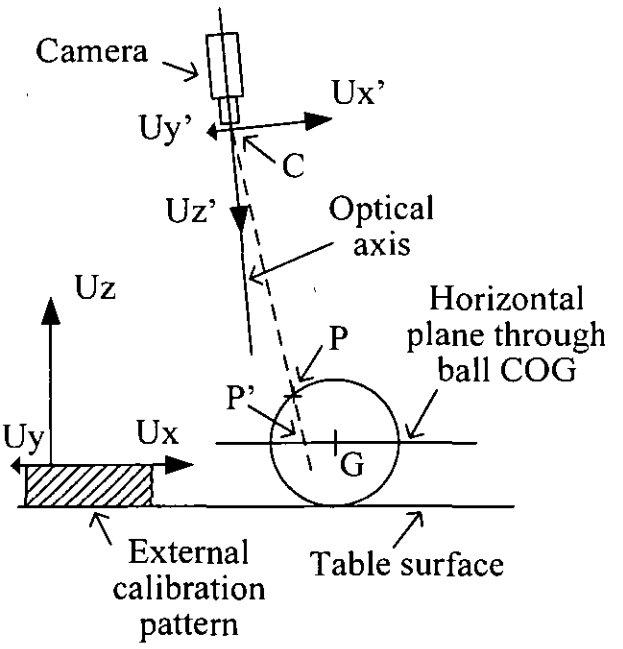


Fig. 5.18. Coordinate systems under consideration for the tracking of a pattern on the ball (this figure must be compared with Figure 4.18)

When the pattern is kept on the upper hemisphere of the ball, it will appear in the images captured by the overhead camera, as shown in Figure 5.17. As in the tracking of the ball, the image-processing algorithm makes all of its spatial measurements on the horizontal plane that goes through the centre of the ball (Figure 5.22). As shown in Figure 5.22, for a pattern position P on the ball surface, the tracking algorithm

calculates the location of the projected line CP onto the horizontal plane that passes through the centre of gravity G of the ball (where C is the origin of the camera coordinate system as given in Figure 4.18). The projected point on the plane is denoted by P'. Alternatively, the image-processing algorithm can be modified to do the projection onto the horizontal surface that goes through P and, to do this, the height of the pattern from the table surface is needed. However, the height of the pattern from the table surface usually changes and is difficult to measure, as the ball is randomly placed on the table, and rotated by hand in order to spin like a top (see Figure 5.19). Once the ball starts to spin, the hand is taken away exposing the ball to the camera, which has already been triggered. Hence, the camera tracks the ball and the pattern.

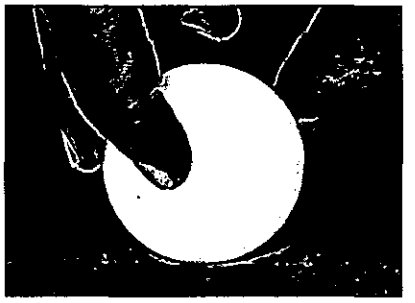


Fig. 5.19. Rotating the ball about the vertical

Hence, the spatial measurements resulting from the image processing algorithm are in fact those of P' and must not be used directly to calculate the value of the ball spin. However, the coordinates of P', when combined with ball radius R, can be helpful in finding the x and y coordinates of P. The process of obtaining the position of P is shown below.

The coordinates of C (the origin of the camera coordinate system Ux'Uy'Uz') in the real-world coordinate system UxUyUz attached to the external calibration pattern can be calculated using the equation $[X']=[R_C]*[X]+[T_C]$ (Section 4.3.2 outlines these concepts). The coordinates of point C (which is the origin of the Ux'Uy'Uz' system) in the UxUyUz axis system are,

$$\begin{bmatrix} x_C \\ y_C \\ z_C \end{bmatrix} = -[R_C]^{-1}[T_C] \tag{5.4}$$

Both the matrices, $[R_C]$ and $[T_C]$ are obtained from the external camera calibration procedure described in Section 4.3.2.

Let $\{x_B, y_B, z_B\}$ be the coordinates of the COG of the ball. Therefore, any general point $\{x, y, z\}$ on the surface of the ball must satisfy the relationship:

$$(x - x_B)^2 + (y - y_B)^2 + (z - z_B)^2 = R^2 \tag{5.5}$$

Now, the equation of the straight-line CPP', in the UxUyUz, with the coordinates of P' being $\{x_{P'}, y_{P'}, z_{P'}\}$, is,

$$\frac{x - x_{P'}}{x_C - x_{P'}} = \frac{y - y_{P'}}{y_C - y_{P'}} = \frac{z - z_{P'}}{z_C - z_{P'}} \tag{5.6}$$

In equation (5.6), the values of $\{x_C, y_C, z_C\}$ are obtained from equation (5.4); also, $z_{P'} = z_B$.

Intersection of the ball's surface and the line CPP' would lead to calculating the coordinate of P, $\{x_P, y_P, z_P\}$, of which x_P and y_P are needed for the estimation of sidespin. When equations (5.5) and (5.6) are solved, there will be two possible coordinates for P, as the roots of a quadratic equation, and the required solution is where $z_P > z_B$ (i.e. where the pattern is at a higher elevation from the table than the ball COG). The required root is:

$$z_P = z_B + R\frac{z_C - z_{P'}}{\sqrt{(x_C - x_{P'})^2 + (y_C - y_{P'})^2 + (z_C - z_{P'})^2}} \tag{5.7}$$

Using (5.4) and (5.7), and also noting that $z_{P'} = z_B$, the value of z_P can be calculated. Then using equation (5.6), x_P and y_P are estimated, using the following equation.

$$\frac{x_P - x_{P'}}{x_C - x_{P'}} = \frac{y_P - y_{P'}}{y_C - y_{P'}} = \frac{z_P - z_{P'}}{z_C - z_{P'}} \tag{5.8}$$

A note on the ball centroid tracking:

When tracking the centroid of the ball, the ball pixels are extracted by the image-processing program initially. Now, irrespective of the camera inclination and the position of the ball on the table, a circular blob will be imaged by the camera. When there is no illumination at all to the lower hemisphere of the ball, the circular blob will not be seen. The camera will image the regions in the lower hemisphere close the horizontal great circle, either if the camera principal axis is inclined largely to the vertical (this is not the case for the current setup) or when the ball is situated farther away from the principal axis. However, such a situation does not arise in the present table area, as several strip lights fitted with standard diffusers illuminate the table and these lights are spread out over a large area around the table. Hence, for all locations of the ball on the table, the actual COG of the ball, G will be detected as the centroid of the circular blob representing the ball in the image. Now the projection of CG on the horizontal plane going though G will be the point G itself. Therefore the calculations performed by equations (5.5), (5.6) and (5.7) are not necessary.

Now for measuring the sidespin of the ball, when the overhead camera tracks a single pattern P put on the ball on its top hemisphere, as depicted in Figure 5.20, the value of the ball sidespin can be calculated as follows:

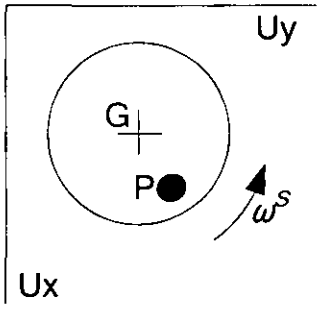


Fig. 5.20. A circular pattern on the ball for spin-tracking

Referring to Figure 5.20, let the tracked coordinates of the pattern P, estimated from equation (5.8) in the i^{th} image frame be $[x_P(i), y_P(i)]$. With $[x_G(i), y_G(i)]$ denoting the coordinates of the ball centroid, sidespin is given by,

$$\omega^S(i) = \left\{ tan^{-1}\left[\frac{y_P(i+1) - y_G(i+1)}{x_P(i+1) - x_G(i+1)}\right] - tan^{-1}\left[\frac{y_P(i) - y_G(i)}{x_P(i) - x_G(i)}\right]\right\} * fps \qquad (5.9)$$

Since the ball is at a stationary position, for every i, $y_G(i) = y_G(i+1)$ and $x_G(i) = x_G(i+1)$.

For sidespin measurements, the ball was spun by hand like a top, as shown in Figure 5.19, such that it does not move laterally on the table. As long as the ball does not move horizontally, it can only spin about the vertical (i.e. it can only have sidespin); this 'zero horizontal movement of the ball' criterion is used to select the right image sequences in order to analyse the variation of sidespin. In addition, the ball was rotated such that the pattern will be in the view of the camera throughout its motion. Moreover, the ball was also kept very close to the optical axis of the camera (Figures 4.17, 5.18) on the table, in order to prevent the inaccuracies that originate from the inclination of the line of sight to the optical axis and the associated problems of there being no illumination on the lower hemisphere of the ball. Ideally, the tracked pattern in the image must make a circular orbit around the centre of the ball.

The captured image appears as seen in Figure 5.17. After an initial image thresholding procedure, in each binary image, the pattern centroid was detected by the MATLAB® Image Processing Toolbox functions *bwlabel* and *regionprops*. Then, the zero intensity (black coloured) pixels of the pattern were eliminated from the ball area by using another function called *imfill*. Now the ball would look as though it had no pattern on it. The centroid of the ball was detected from the 'filled' image. When the continuously-tracked coordinates of the ball and the pattern were superimposed on the first image of the sequence, the plot obtained is similar to Figure 5.21. It should be noted that the tracked path of the pattern, instead of being around the centre of the ball, is rather shifted towards the lower-left hand corner of the image, as shown in Figure 5.21. When the individual grey-scale images in the sequence were examined with the naked eye, an extra glare was found to occur on the pattern as it approached the upper-right hand side of the image shown in Figure 5.21. However, due to a strip light source (a fluorescent lamp) located right above the table area, on the upper-right side of Figure 5.19, there was extra light reflection (and associated illumination) on the black-coloured pattern when it was at that side of the ball centre.

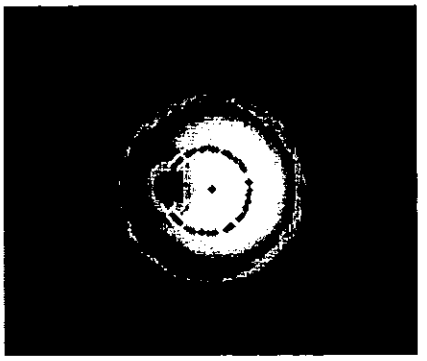


Fig. 5.21. Tracking a spinning ball – grey scale image (multiple complete rotations of the ball captured at 180fps)

Equation (5.9) was used to calculate the sidespin of the ball. An angular speed-time plot for this ball-tracking is shown in Figure 5.22. The waviness in the scatter of points is due to the effect of orbit shift as described earlier. As seen in Figure 5.22, for the angular velocity variation, the period of oscillation increases with time, also, the fluctuation decreases. These two effects are due to the progressive reduction of the ball velocity with time, so that the ball takes a longer time to complete a rotation as time elapses. A best-fit straight line gives an average angular deceleration of 21.8 rad/s^2 for the spinning ball. The ball was tracked several times and the deceleration of the ball was found to be in the range of 22 rad/s^2, which is almost 3.5 rotations/s^2.

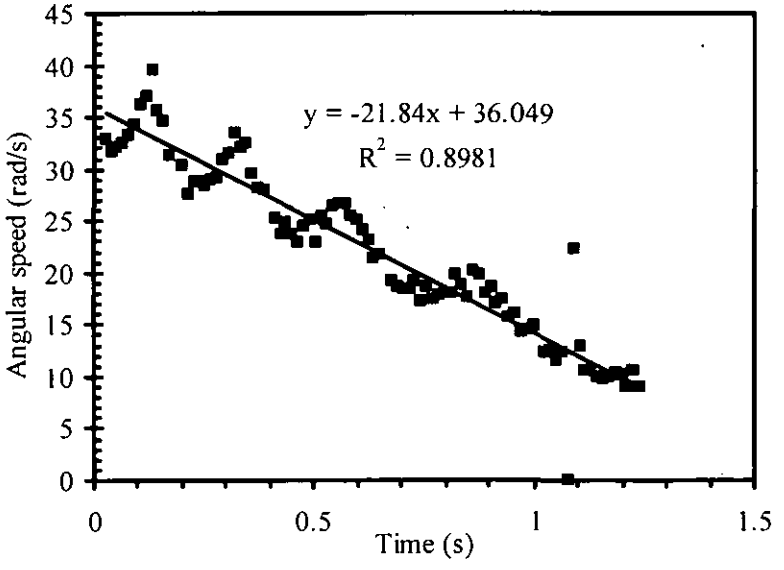


Fig. 5.22. Sidespin (angular speed) vs. time plot for a ball spinning from a stationary position

5.4 A Theory of 3-Dimensional Spin Tracking Using a Single Pattern

A single circular pattern is also intended to be used to track the spin here, as in Section 5.3 (see Figure 5.21). Since the cue ball will be moving on the table, unlike the situation described in Section 5.3, a requirement to track the ball spin is that the pattern must not disappear for the first few frames after cueing (i.e. the pattern must be on the upper hemisphere of the ball in order to be captured by the camera). Hence, this method can only be applied to track the ball spin within a limited area on the table. However, this procedure is expected to be very useful in determining the cueing dynamics, where only the initial spin of the ball, immediately after cueing, is of interest.

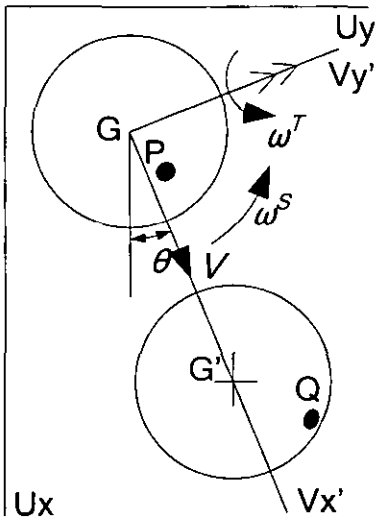


Fig. 5.23 Tracking topspin and sidespin

Figure 5.23 depicts a typical scenario of ball motion. Initially, the ball is at G and the pattern centroid is depicted as P on the ball. For the next image frame, let the ball be at G' and the new position of the pattern be Q. Both P and Q denote the vertical projections of the pattern positions on the ball (i.e. the coordinates of P and Q are those obtained from equation (5.8)). GG' is the direction of the ball motion, which is at θ to the Ux axis. Unless the cue inclination is very high, which produces a curving effect on the ball, θ remains a constant for a shot when all of its image frames are considered. Axis system Vx'Vy' is selected as it has G as its origin, and Vx' is along the direction of the ball velocity V. The absolute movement of the pattern from P to Q

is due to the contribution of the ball's linear speed, sidespin and topspin V, ω^S and ω^T respectively. When the pattern's movement relative to the centroid of the ball is considered, the effect of V on the positional change of the pattern need not be taken into account. Also, throughout this chapter V can be estimated from the length GG'. Here, the movement of the pattern relative to the ball centroid is used to estimate the instantaneous values of sidespin and topspin, ω^S and ω^T, respectively.

Figure 5.24 shows the relative movement of the pattern to that of the ball centroid, hence both G and G' are now represented by the same point in the figure. Only a part of the ball is shown in Figure 5.25.

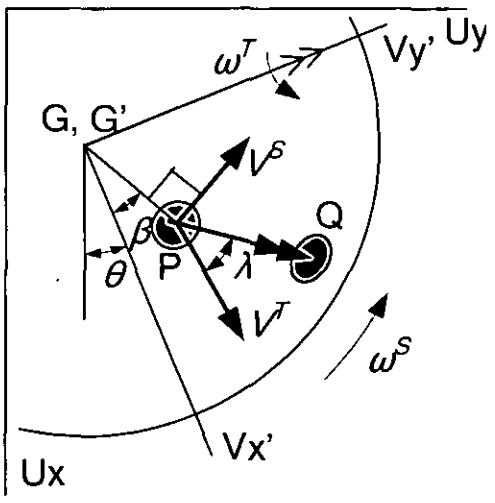


Fig. 5.24 Movement of the pattern relative to ball centroid (spatial locations and velocities are projected onto the horizontal plane that goes through the centre of ball)

In Figure 5.24, the movement from P to Q is caused by two velocity components: V^S due to sidespin ω^S, and V^T from topspin ω^T. V^S will always be parallel to the plane shown in the figure. V^T in the figure is the horizontal component of velocity acting on the pattern due to topspin ω^T (also see Figure 5.25). V^S is perpendicular in direction to the line GP and its direction is determined by the direction of ω^S (i.e. whether it is right spin or left spin). V^T is parallel to the Vx' axis and its direction is decided by the sign of ω^T. (Topspin is considered positive, while bottom spin is treated as negative).

The shorter the length of PQ, the more accurate the motion of the pattern described by the forgoing model. Conversely, if PQ is long, the distance covered due to ω^S will not be on a straight line along the direction of V^S as shown in Figure 5.24, but will rather be described by a circular arc having G as its centre. The length of PQ can be minimised by increasing the frame rate of the camera (fps). The length of PQ is greatly exaggerated with respect to the ball size in Figure 5.24, for the purposes of clarity.

Let G and P belong to the image frame i and say G' and Q are on the image frame i+1. Also using the notation that was used in Equation (5.9), i.e. treating Q as P(i+1), angles θ, β, λ in Figure 5.24 are determined by,

$$\theta(i) = tan^{-1}\left(\frac{y_G(i+1) - y_G(i)}{x_G(i+1) - x_G(i)}\right) \tag{5.10}$$

$$\beta(i) = tan^{-1}\left(\frac{y_P(i) - y_G(i)}{x_P(i) - x_G(i)}\right) - \theta(i) \tag{5.11}$$

$$\lambda(i) = tan^{-1}\left\{\frac{[y_P(i+1) - y_G(i+1)] - [y_P(i) - y_G(i)]}{[x_P(i+1) - x_G(i+1)] - [x_P(i) - x_G(i)]}\right\} - \beta(i) - \theta(i) \tag{5.12}$$

It must be noted that in equations (5.10), (5.11), and (5.12), all x and y coordinates are the ones obtained from equation (5.8).

Taking Δt as the time elapsed between frames i and i+1, i.e. $\Delta t = \frac{1}{fps}$, from the velocity vectors shown in Figure 5.25, resolving the distances along the Vy' axis,

$V^S(i)\Delta t \cos\beta(i) = PQ(i)\sin\lambda(i)$.

Also, $V^S(i) = \omega^S(i)GP(i)$. Note the length segment denoted as GP in Figure 5.25 as well. Now,

$$\omega^S(i) = \frac{PQ(i)\sin\lambda(i)}{GP(i)\cos\beta(i)}\, fps \tag{5.13}$$

where,

$$GP(i) = \sqrt{\{x_P(i) - x_G(i)\}^2 + \{y_P(i) - y_G(i)\}^2} \tag{5.14}$$

and,

$$PQ(i) = \sqrt{\{[x_P(i+1) - x_G(i+1)] - [x_P(i) - x_G(i)]\}^2 + \{[y_P(i+1) - y_G(i+1)] - [y_P(i) - y_G(i)]\}^2}\,. \tag{5.15}$$

Considering the 3-dimensional position of point P on the ball, as shown in Figure 5.25, it can be easily shown that, $V^T(i) = \omega^T(i)[h_P(i) - R]$, where h_P is the height of the pattern on the ball from the table plane, and R is the ball radius. With reference to Figure 5.25, it can also be shown that, $h_P(i) = R + \sqrt{R^2 - [GP(i)]^2}$.

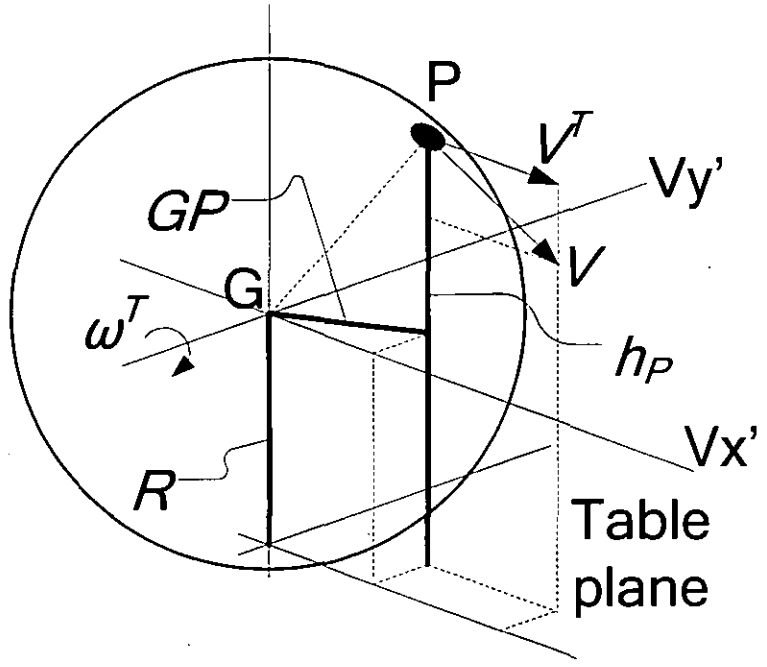


Fig. 5.25. Velocity of the ball due to topspin

From the velocity vectors shown in Figure 5.25, resolving along the Vx' axis, $[V^T(i) - V^S(i)\sin\beta(i)]\Delta t = PQ(i)\cos\lambda(i)$ and, when combined with the above two equations, this becomes,

$$\omega^T(i) = \frac{PQ(i)\cos\lambda(i) * fps + V^S(i)\sin\beta(i)}{\sqrt{R^2 - [GP(i)]^2}}$$

Also making use of equation (5.13), this can be further simplified to,

$$\omega^T(i) = [\cos\lambda(i) + \sin\lambda(i)\tan\beta(i)]\frac{PQ(i) * fps}{\sqrt{R^2 - [GP(i)]^2}} \qquad (5.16)$$

Equations (5.13) and (5.16) will be used in Chapter 7 to estimate the sidespin and the topspin of the ball immediately after cueing, in order to establish an empirical model for the cueing dynamics.

Summary

High-speed video capture using a single machine-vision camera was found to give good results in determining the dynamics involved in snooker. The rolling and sliding coefficients of frictions have been found. In relation to the collisions found in snooker some experiments were carried out under controlled experimental conditions, which enables the calculation of the parameter values that influence the impact dynamics in the next chapter. The snooker cue was also tracked using the overhead camera and by using this, human and robot cueing performances were compared. The cue-embedded force sensor was also used to measure the forces present during cueing. The cue tracking and the force sensor results were used to conclude that force sensor readings are redundant when it comes to the robot's decision-making about a specific shot and that only the speed of the cue launcher is important. Using a single circular pattern, the ball sidespin was tracked to determine the resistance of the table to sidespin. However, it was found that inconsistent illumination conditions over the table area affect the algorithms that are used for spin tracking. A model for determining topspin and sidespins after the cueing using only a single circular pattern is also presented.

Chapter 6

ANALYSIS OF COLLISIONS AND BALL TRAJECTORIES

It was emphasized in Chapter 2 that potting accuracy and the positional play of the cue ball are the two primary skills that professional players rely on. Both these phenomena involve estimating the ball trajectories. According to the literature review, the trajectories of a ball can be described analytically, but no complete models exist for the collisions that are encountered in snooker and pool. In this chapter, two types of collisions, one between two snooker balls and the other between a ball and a cushion, are analysed using the principles of impact mechanics. This model-based information will, ultimately, lead to a more intelligent decision-making in the robotic snooker playing system. Understanding derived from this study will also add to the wider knowledge base in snooker and frictional impacts. In Section 6.1, an analysis of the post-collision trajectories of two balls is presented. Section 6.2 provides a theoretical analysis to obtain the trajectories after a ball-cushion impact.

6.1 Frictional Collisions between the Balls

Here, the problem involves the cue ball, C, obliquely impinging onto another object ball O (see Figure 6.1). Both the cue and object balls are of equal mass and radius. Traditionally, the ball collisions are analyzed without incorporating the effect of friction, and the object ball is supposed to move along the line connecting the ball centres at the instant of impact **[Wallace and Schroeder 1988]**. However, when a ball is spinning and colliding into another, as shown in Figure 6.1, in addition to the normal forces that are usually set up between them, frictional forces are also introduced. These forces drastically change the ball trajectories. This effect is called *throw* in billiards. Few researchers have considered this effect to derive a solution for the collision problem and to obtain the amount of throw, which is the deviation from the ideal direction without any friction **[Marlow 1994, Alciatore 2008]**.

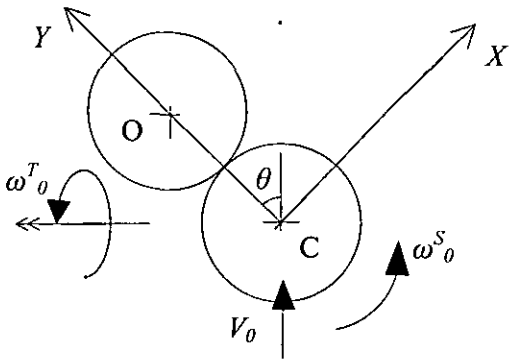


Fig.6.1. Oblique impact of spheres (direction of ω^T_0 is given by the right hand grip rule)

In snooker/billiards, the balls are also rolling on a frictional surface (i.e. the table). In such cases, frictional forces from the surface upon which they move also act on the balls **[de la Torre Juarez 1994]**. Marlow **[1994]** also acknowledges the effect of surface friction on the impact between the balls, but has not obtained any solutions along these lines. In a very recent paper, Domenech **[2008]** has tried to address the issue of surface friction influencing the impact between balls. However, Domenech assumes that the slip is uni-directional throughout the impact. This assumption may be true for very low friction bodies. However, the approach that is taken in the current work provides a generic solution that can be applicable for any spheres irrespective of their friction coefficients. The generic procedure as provided in this section can lead to the identification of conditions under which the assumption of uni-directional slip can be used.

In the following analysis, initially a general solution will be derived for the problem of two identical balls colliding obliquely and, at the end, the values applicable for snooker will be substituted. In Figure 6.1, it is important to note that ball C does not spin about its frontal axis (about the direction of V_0), this condition is only prevalent during a massé shot and is not normally encountered in billiards.

When two spheres collide, they are generally treated as deformable bodies; hence the contact between them is made over a region rather than at a point. The contact area between the spheres during impact is usually estimated through the Hertz theory. Researchers have also used various Finite Element Analysis techniques to analyse the

contact area between the spheres **[Zhang and Vu-Quoc 2002]**. However, to calculate the deformation at the interface of the spheres, a number of material properties such as the Poisson ratio and Young's modulus are needed. In the absence of such material properties for snooker balls, a point contact is assumed between the balls during impact. In addition, snooker balls are quite rigid and when the Young's moduli of the spheres are high, i.e. the spheres are less deformable, the Hertz theory predicts a small deformation. The assumption of a point contact has also been used by other researchers like Domenech **[2008]**.

6.1.1 General Equations of Motion

In Figure 6.2, for sphere C, for the linear motion along X, Y and Z directions,

$$F_1 + F_{R,x}^C = M\,\ddot{x}_G^C \tag{6.1a}$$

$$-F_1 + F_{R,y}^C = M\,\ddot{y}_G^C \tag{6.1b}$$

$$F_2 + F_N^C - mg = M\,\ddot{z}_G^C \tag{6.1c}$$

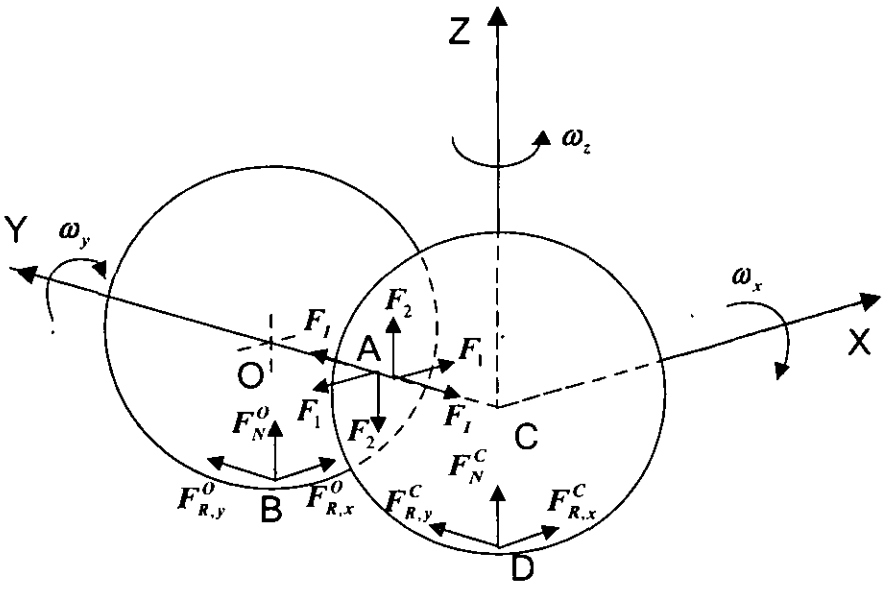


Fig. 6.2. The forces acting on the balls during the impact

During collision, at any time instant t, consider a time period of Δt. Now, let ΔP denote the impulse due to the action of a general force F over Δt. Also the

accumulated total impact up to time t is denoted as P (also assuming that the impact started at t=0), now, it can be written that,

$$\Delta P = \int_{t}^{t+\Delta t} F.dt \tag{6.2a}$$

and,

$$P = \sum \Delta P = \int_{0}^{t} F.dt \tag{6.2b}$$

For sphere C, at time t during the collision, consider an increment Δt in time, from the equations in (6.1) and (6.2)

$$\Delta P_1 + \Delta P_x^C = M\left[\dot{x}_G^C(t+\Delta t) - \dot{x}_G^C(t)\right] \tag{6.3a}$$

$$-\Delta P_1 + \Delta P_y^C = M\left[\dot{y}_G^C(t+\Delta t) - \dot{y}_G^C(t)\right] \tag{6.3b}$$

$$\Delta P_2 + \Delta P_N^C = M\left[\dot{z}_G^C(t+\Delta t) - \dot{z}_G^C(t)\right] \tag{6.3c}$$

In equation (6.3c), it should be noted that the impact component due to the mass of the ball Mg is absent. According to de la Torre Juarez **[1994]**, in the limit $\Delta t \rightarrow 0$, the non-diverging forces, such as the weight Mg, will have a negligible contribution and thus will not influence the increase in momentum. A simple calculation also confirms this fact. When, say, 1m/s speed was transferred from the cue ball to the object ball (this is a typical average speed), the collision time was measured to be 300 μs, by the set-up given in Figure 4.4 **[Marlow 1994]**. For a ball mass of around 140 g, the average impulse force would be 470 N, whereas the weight of the ball is only 1.4 N.

The moment of impulse and the angular momentum about the centre of mass for the ball C about X, Y and Z directions, are given by,

$$\left(\Delta P_2 + \Delta P_y^C\right)R = \frac{2MR^2}{5}\left[\omega_x^C(t+\Delta t) - \omega_x^C(t)\right] \tag{6.4a}$$

$$-\Delta P_x^C R = \frac{2MR^2}{5}\left[\omega_y^C(t+\Delta t) - \omega_y^C(t)\right] \tag{6.4b}$$

$$-\Delta P_1 R = \frac{2MR^2}{5}\left[\omega_z^C(t+\Delta t) - \omega_z^C(t)\right] \tag{6.4c}$$

Similarly for ball O,

$$-\Delta P_I + \Delta P_x^O = M\left[\dot{x}_G^O(t+\Delta t) - \dot{x}_G^O(t)\right] \tag{6.5a}$$

$$\Delta P_I + \Delta P_y^O = M\left[\dot{y}_G^O(t+\Delta t) - \dot{y}_G^O(t)\right] \tag{6.5b}$$

$$-P_2 + P_N^O = M\left[\dot{z}_G^O(t+\Delta t) - \dot{z}_G^O(t)\right] \tag{6.5c}$$

$$\left(\Delta P_2 + \Delta P_y^O\right)R = \frac{2MR^2}{5}\left[\omega_x^O(t+\Delta t) - \omega_x^O(t)\right] \tag{6.6a}$$

$$-\Delta P_x^O R = \frac{2MR^2}{5}\left[\omega_y^O(t+\Delta t) - \omega_y^O(t)\right] \tag{6.6b}$$

$$-\Delta P_I R = \frac{2MR^2}{5}\left[\omega_z^O(t+\Delta t) - \omega_z^O(t)\right] \tag{6.6c}$$

These equations are adequate to describe the change in the motion of the balls due to the impact.

6.1.2 Impact Dynamics

At the contact point between the spheres, i.e. at A, let the relative speed of ball C to that of ball O be $s(t)$ at an angle $\Phi(t)$ with the X-axis (the relative velocity vector will lie on the XZ plane). The instantaneous value of the normal impulse P_I (which will be the accumulated value of all ΔP_I's until time t) is always positive within the interval of impact. In addition, P_I monotonously increases with time, thus, in this analysis it is taken as an independent variable instead of the usual variable of time t **[Stronge 2000]**. Slipping velocities along the X and Z axes respectively,

$$\dot{x}_A = \dot{x}_A^C - \dot{x}_A^O = s(P_I)\cos(\Phi(P_I)) \tag{6.7a}$$

$$\dot{z}_A = \dot{z}_A^C - \dot{z}_A^O = s(P_I)\sin(\Phi(P_I)) \tag{6.7b}$$

The normal component of relative velocity,

$$\dot{y}_A = \dot{y}_A^C - \dot{y}_A^O \tag{6.7c}$$

For the nominal slipping speeds to be along the positive X and Z axes, when the balls are sliding on each other at their contact point A, from the Amontons-Coulomb law,

$$\Delta P_1 = -\mu_{bb} \cos(\Phi(P_I))\Delta P_I \tag{6.8a}$$

$$\Delta P_2 = -\mu_{bb} \sin(\Phi(P_I))\Delta P_I \tag{6.8b}$$

where μ_{bb} is the coefficient of sliding friction between the spheres.

Since ball O is moving on the plane, to satisfy the condition $\dot{z}_G^O(t+\Delta t) - \dot{z}_G^O(t) = 0$, and from (6.5c),

$$\Delta P_N^O = \Delta P_2 = -\mu_{bb} \sin(\Phi(P_I))\Delta P_I \tag{6.9}$$

Similarly, for ball O, at B slip s' and its direction Φ' with the X axis (s' will on the XY plane), for the ball to slide, and also using (6.9),

$$\Delta P_x^O = -\mu_s \cos(\Phi')\Delta P_N^O = \mu_{bb}\mu_s \sin(\Phi(P_I))\cos(\Phi'(P_I))\Delta P_I \tag{6.10a}$$

$$\Delta P_y^O = -\mu_s \sin(\Phi')\Delta P_N^O = \mu_{bb}\mu_s \sin(\Phi(P_I))\sin(\Phi'(P_I))\Delta P_I \tag{6.10b}$$

Here μ_s is the sliding friction coefficient between the ball and the table, measured to be 0.21 in Chapter 5. Sliding speeds are:

$$\dot{x}_B = \dot{x}_B^O = s'(P_I)\cos(\Phi'(P_I)) \tag{6.11a}$$

$$\dot{y}_B = \dot{y}_B^O = s'(P_I)\sin(\Phi'(P_I)) \tag{6.11b}$$

From now onwards the independent variable P_I is omitted in the equations in order to keep them compact.

Notably depending on the value of $\dot{z}_A$ (as given in equation (6.7b)), some of the impulses in the equation sets (6.3), (6.4), (6.5) or (6.6) will be zero.

If $\dot{z}_A$ is negative, the cue ball will have more downward velocity (along the Z axis) at the contact point A, and the frictional impulse ΔP_2 between the balls will be acting on the balls in the directions as shown in Figure 6.2 (i.e. $\Delta P_2 > 0$). If ball C is to remain on the table, from equation (6.3c), $\Delta P_2 + \Delta P_N^C < 0$. These two conditions can be satisfied only when $\Delta P_N^C < 0$, and apparently it is impossible to satisfy this condition as the table cannot apply a 'negative' reaction on the ball. Thus, $\Delta P_N^C = 0$, which in turn allows us

to say that the associated frictional impulses are also absent, i.e. $\Delta P_x^C = 0$ and $\Delta P_y^C = 0$. Here, ball C will lift up from the table, like a cue ball in a 'jump' shot, however small the lift is. However, it is assumed that during the time of the impulse it remains at the same spatial location, just above the table, without altering the configuration given in Figure 6.2. This assumption is reasonable since the time of impulse between two balls is very small and is in the range of 0.3 ms **[Marlow 1994]**. Conversely, if $\dot{z}_A$ is positive, then $\Delta P_N^O = 0$, $\Delta P_x^O = 0$ and $\Delta P_y^O = 0$.

6.1.2.1 Conditions for Rolling

When rolling occurs, slip speed s (or s' for the sliding on the table), becomes zero. At this instance the relative motion between bodies stops at their contact point along the common tangent. Neglecting the effects of static friction, the frictional forces are assumed zero.

1) When $s=0$, which is a common occurrence depending on the initial conditions, as shall be seen shortly, where both the spheres will be rolling on each other at their contact point A.

$\Delta P_1 = \Delta P_2 = 0$, that also follows $\Delta P_N^O = \Delta P_x^O = \Delta P_y^O = 0$

2) When $s'=0$, $\Delta P_x^O = \Delta P_y^O = 0$, and the sphere O will roll on the plane without sliding.

6.1.2.2 Coefficient of Restitution

According to Stronge **[2000]**, the energetic coefficient of restitution e, is independent of friction and the process of slip. e^2 is the negative of the ratio of the work done by the impulse force during the phase restitution to that during the compression phase. When P_I^f, P_I^c denote the accumulated impulse at the termination of the impulse, and at the termination of the compression, respectively, it is possible to show that the work done is

$$\Delta W_y = \int F_I . \dot{y}_A dt = \int \dot{y}_A dP_I \qquad (6.12a)$$

$$e^2 = \frac{-\int_{P_I^c}^{P_I^f} \dot{y}_A dP_I}{\int_0^{P_I^c} \dot{y}_A dP_I}$$

This can be rearranged as

$$W_y\left(P_I^f\right) = \left(1 - e^2\right) W_y\left(P_I^c\right) \tag{6.12b}$$

The termination of compression occurs when the normal component of relative velocity becomes zero, i.e.,

$$\dot{y}_A\left(P_I^c\right) = 0 \tag{6.12c}$$

6.1.3 Velocity Relationships

The velocity of any point on a sphere's surface can be written in vector notation as,

$\vec{V} = \vec{V}_G + \vec{\omega} \Lambda \vec{R}$

At A,

$$\dot{x}_A^C = \dot{x}_G^C - R\omega_z^C,\ \dot{z}_A^C = \dot{z}_G^C + R\omega_x^C,\ \dot{x}_A^O = \dot{x}_G^O + R\omega_z^O \text{ and } \dot{z}_A^O = \dot{z}_G^O - R\omega_x^O \tag{6.13a}$$

At B,

$$\dot{x}_B^O = \dot{x}_G^O - R\omega_y^O \text{ and } \dot{y}_B^O = \dot{y}_G^O + R\omega_x^O \tag{6.13b}$$

6.1.4 Solution for the ball velocities

Sets of equations in sections 6.1.1, 6.1.2 and 6.1.3 allow the calculation of the increments in the centroid velocities of balls C and O, $\{\Delta\dot{x}_G^C, \ldots, \Delta\omega_z^C\}$ and $\{\Delta\dot{x}_G^O, \ldots, \Delta\omega_z^O\}$ respectively, as given in equation sets (6.3), (6.4), (6.5) and (6.6). For example, for $\Delta\dot{x}_G^C$,

from equations (6.3a), (6.8a) and with the assumption that $\Delta P_x^C = 0$,

$$\Delta\dot{x}_G^C = -\frac{\mu_{bb}}{M} \cos\Phi \Delta P_I \tag{6.14a}$$

Φ can also be expressed in terms of the centroid velocities of the balls using equations (6.7a), (6.7b) and (6.12a), as

$$\cos\Phi = \frac{\dot{x}_A}{\sqrt{(\dot{x}_A)^2 + (\dot{z}_A)^2}}$$

$$\cos\Phi = \frac{\dot{x}_G^C - R\omega_z^C - \dot{x}_G^O - R\omega_z^O}{\sqrt{\left(\dot{x}_G^C - R\omega_z^C - \dot{x}_G^O - R\omega_z^O\right)^2 + \left(\dot{z}_G^C + R\omega_x^C - \dot{z}_G^O + R\omega_z^O\right)^2}}$$

Now (6.13a) becomes,

$$\Delta\dot{x}_G^C = -\frac{\mu_{bb}}{M} \frac{\dot{x}_G^C - R\omega_z^C - \dot{x}_G^O - R\omega_z^O}{\sqrt{\left(\dot{x}_G^C - R\omega_z^C - \dot{x}_G^O - R\omega_z^O\right)^2 + \left(\dot{z}_G^C + R\omega_x^C - \dot{z}_G^O + R\omega_z^O\right)^2}} \Delta P_I \tag{6.14b}$$

This can be expressed as a differential equation,

$$\frac{d\dot{x}_G^C}{dP_I} = -\frac{\mu_{bb}}{M} \frac{\dot{x}_G^C - R\omega_z^C - \dot{x}_G^O - R\omega_z^O}{\sqrt{\left(\dot{x}_G^C - R\omega_z^C - \dot{x}_G^O - R\omega_z^O\right)^2 + \left(\dot{z}_G^C + R\omega_x^C - \dot{z}_G^O + R\omega_z^O\right)^2}} \tag{6.14c}$$

Deriving similar expressions for the other 11 components of the centroid velocities of both balls will result in 12, simultaneous, nonlinear, differential equations. Exact solutions are not available. A numerical solution is possible. For, example for the problem of rotating spheres colliding with each other (like a mid-air collision), Kane and Levinson **[1987]** have used a numerical scheme to obtain the variation of the sliding velocities, etc during the time of impact, for both the spheres. When it comes to robotic nonprehensile manipulation, Li and Payandeh **[2003]** modelled the trajectory and orientation of a polygonal, flat plate using numerical simulations. In order to perform a numerical operation, equation (6.13c) must be written in numerical form as:

$$\left(\dot{x}_G^C\right)_{n+1} - \left(\dot{x}_G^C\right)_n$$

$$= -\frac{\mu_{bb}}{M} \frac{\left(\dot{x}_G^C\right)_n - R\left(\omega_z^C\right)_n - \left(\dot{x}_G^O\right)_n - R\left(\omega_z^O\right)_n}{\sqrt{\left[\left(\dot{x}_G^C\right)_n - R\left(\omega_z^C\right)_n - \left(\dot{x}_G^O\right)_n - R\left(\omega_z^O\right)_n\right]^2 + \left[\left(\dot{z}_G^C\right)_n + R\left(\omega_x^C\right)_n - \left(\dot{z}_G^O\right)_n + R\left(\omega_z^O\right)_n\right]^2}} \Delta P_I$$

(6.15a)

The work done during an increment of ΔP_I can also be calculated numerically, using equation (6.12 a),

$$\left(W_y\right)_{n+1} - \left(W_y\right)_n = \Delta P_I \frac{\left[\left(\dot{y}_A\right)_{n+1} + \left(\dot{y}_A\right)_n\right]}{2} \tag{6.16}$$

6.1.4.1 Initial Conditions

When P_I=0,

For ball C, $\left(\dot{x}_G^C\right)_1 = V_0 \sin\theta$, $\left(\dot{y}_G^C\right)_1 = V_0 \cos\theta$, $\left(\dot{z}_G^C\right)_1 = 0$, $\left(\omega_x^C\right)_1 = -\omega_0^T \cos\theta$,

$\left(\omega_y^C\right)_1 = \omega_0^T \sin\theta$, $\left(\omega_z^C\right)_1 = \omega_0^S$.

For ball O, $\left(\dot{x}_G^O\right)_1 = 0$, $\left(\dot{y}_G^O\right)_1 = 0$, $\left(\dot{z}_G^O\right)_1 = 0$, $\left(\omega_x^O\right)_1 = 0$, $\left(\omega_y^O\right)_1 = 0$, $\left(\omega_z^O\right)_1 = 0$.

Also, $s(0) = \left|\sqrt{\left(V_0 \sin\theta - R\omega_0^S\right)^2 + \left(R\omega_0^T \cos\theta\right)^2}\right|$ and $s'(0) = \left|V_0 - R\omega_0^T\right|$.

6.1.4.2 Numerical Algorithm

The numerical scheme is written in MATLAB® programming language. The values of V_0, ω_0^T, ω_0^S, θ are the inputs to the scheme. The smaller the value of the increment in impulse P_I, i.e. ΔP_I, in (6.15a), the more accurate the results will be. The aim is to find the centroid velocities of the balls at the final accumulated impulse value P_I^f.

The code initially starts by calculating the increments in the centroid velocities of both balls by using equation (6.15a) and 11 other simultaneous equations as explained in Section 6.1.4. Using these and equations (6.13a) and (6.13b) the new slip velocities are calculated. The code is designed to incorporate the necessary modifications, as given in Section 6.1.2.1, when a rolling condition is reached at either of the sliding contacts.

Again P_I^f cannot be found analytically and has to be obtained numerically using the equations (6.17a) and (6.17b). Initially P_I^c is calculated, from (6.17b), when the relative speed in the normal direction becomes zero in the numerical scheme i.e. $\dot{y}_A\left(P_I^c\right) = 0$. Until then, the work done is calculated in the scheme using equation (6.16), this enables calculating the total work from the normal impulse, $W_y\left(P_I^c\right)$ and then using (6.12b) the total work at the termination of impulse $W_y\left(P_I^f\right)$ can be estimated. The numerical algorithm is stopped when $W = W_y\left(P_I^f\right)$.

For snooker balls, M=0.1406 kg and R=26.2 mm.

Note: In order to assume a reasonable value for ΔP_I to start the numerical scheme, an approximate value for P_I^f can be taken as $(1+e)MV_0 \cos\theta$, the value of the impulse had the ball O been a solid wall and, say approximately for N iterations, $\Delta P_I = \frac{(1+e)MV_0 \cos\theta}{N}$. Clearly the values of P_I^c and P_I^f will decide the actual number of iterations that have taken place in the scheme.

6.1.4.3 The values of μ_{bb} and e

The values for these parameters presented in the literature are vague with only Marlow [1994] reporting about them. Marlow predicts a value of 0.06 for μ_{bb} for well-polished balls. Furthermore, when the balls have any other substance, such as chalk, on their surface, according to Marlow the value may be as high as 0.2. Also, quite contradictorily, a variation of μ_{bb} in the form of $\mu_{bb} = 9.951 \times 10^{-3} + 0.108e^{-1.088s}$ is also put forward by Marlow **[1994]**, where s denotes the slip speed between the balls. The experimental process in obtaining the aforesaid variation did not seem sufficiently reliable. According to Marlow $e \geq 0.92$. It is believed that the high-speed camera measurements described in Chapter 5 are far superior to the techniques used by Marlow, and therefore these results are used for the calculations and simulations performed. In addition, the adhesive properties between the balls could lead to the phenomenon of stick **[Thronton and Ning 1998]**. Stick between the balls is neglected considering the polished nature of the ball suface. Therefore only the effects of the sliding friction is assumed to be present at the contact point of the spheres. In addition, the coefficient of restitution between the balls is assumed a constant and its minor variations with the impact velocity of the balls, as explored by researchers such as Zhang and Vu-Quoc **[2002]**, are neglected.

The experimental plot shown in Figure 5.12 is used in conjunction with these numerical simulations to obtain the values of the coefficient of restitution and the value of sliding friction. The fundamental idea is to replicate the experimental results by numerical simulations, by using two random numerical values for the above

parameters by a trial-and-error procedure. The plot in Figure 5.12 was obtained under the conditions of $\theta = 0°$, $\omega_0^S = 0$ and $\omega_0^T = \frac{V_0}{R}$. For each of the incident speed values V_0 given in Figure 5.12, the value of the centroid velocity of the object ball at the termination of impact $\dot{y}_G^O(P_I^f)$ was found numerically for e in the range 0.7 to 1.0 and μ_{bb} between 0 and 0.2, both in 0.01 increments. For given values of μ_{bb} and e, the RMS value of all the errors between the experimental and the numerically obtained values for each of the incident velocities given in Figure 5.12 was obtained. The values of μ_{bb}=0.05 and e=0.89 were found to have the least RMS value. The agreement of these values with Marlow's **[1994]** values of 0.06 and 0.92 should be noted. The experimental plot of Figure 5.12 is replicated with the predictions from the numerical scheme in Figure 6.3.

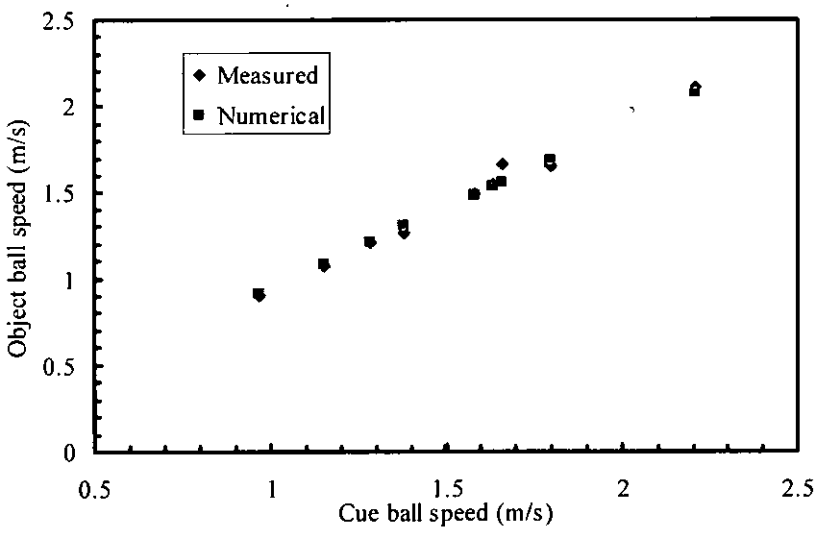


Fig.6.3. Experimental results and numerical predictions for μ_{bb}=0.05 and e=0.89 and under the conditions of $\theta = 0°$, $\omega_0^S = 0$ and $\omega_0^T = \frac{V_0}{R}$

6.1.5 Parabolic Path Subsequent to Impact

When the ball shown in Figure 6.4 spins about its velocity axis (here it has a spin component of ω_1 about its centroid velocity, which in this case is V), irrespective of the other two spin components, the ball will move along a curved path. This is called

massé in billiards. Curved shots can be made by elevating the cue when striking the ball. Curved ball trajectories are also produced due to frictional percussions during the impact between two balls or that between a ball and a cushion. The 2nd type is of interest here. However, the derivation given in this section is, essentially, applicable for any general curved shot.

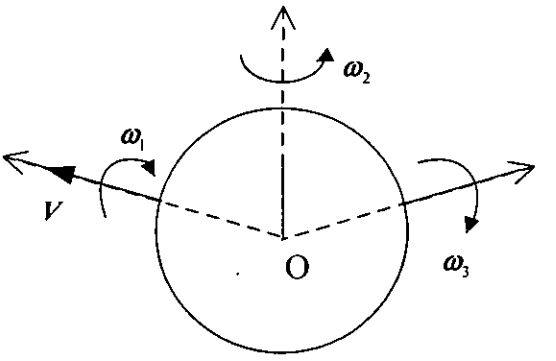


Fig.6.4 A ball that spins about its frontal velocity axis

When the numerical scheme that is described in Section 6.1.4.2 is executed, it is found that generally both the cue ball and the object ball would have spin about their frontal velocity axes. Here the example of the object ball O is used to derive, with appropriate symbols, the description of the trajectory of the ball under massé conditions.

The effect of table friction will generally impart a spin about the velocity axis of the object ball, as in a spinning bullet. This will curve the path of ball O immediately after the termination of impact, making the final direction of movement different to that at the termination of impulse. Its effect will be very prominent for high values of μ_{bb}. In mathematical terms, at the termination of impact, when n=N, i.e. the final step of the numerical algorithm, this condition for curving is created when,

$$\Phi'_N \neq \beta_N \tag{6.17}$$

Where β_N denotes the direction of movement of the centre of gravity of the ball O (i.e. the direction of its velocity), given by $tan\,\beta_N = \frac{(\dot{y}_G^O)_N}{(\dot{x}_G^O)_N}$, and Φ' being its slipping direction on the table. It should be noted that any person will only observe β and not Φ', at any time during the motion of the ball.

The subsequent curved path of the ball can be shown to describe a parabola, conveniently in the $X'Y'$ coordinate system, which is rotated from the XY axes by ψ (see Figure 6.5). Where,

$$tan\psi = -\frac{1}{tan\Phi'_N} \tag{6.18}$$

and Φ'_N is obtained from the numerical algorithm explained earlier. Most of the following equations used below are taken from the detailed analysis of Hopkins and Patterson **[1977]**, where they analyze the curved path of a bowling ball, and modified accordingly.

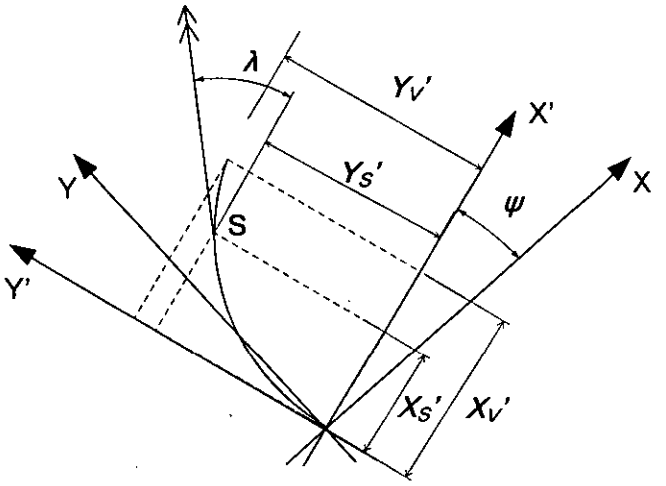


Fig. 6.5. Curved path of O (the object ball)

$$(y'-Y_V') = -\left[\frac{Y_V'}{(X_V')^2}\right](x'-X_V')^2 \tag{6.19}$$

Where,

$X_V' = \frac{V_{X'}(0)V_{Y'}(0)}{\mu_s g}$, $Y_V' = \frac{(V_{Y'}(0))^2}{2\mu_s g}$, and the initial centroid velocities of the ball along X' and Y' are denoted by $V_{X'}(0)$ and $V_{Y'}(0)$. When substituted with the parameters used in this analysis, the following can be written,

$$X_V' = \frac{\left[(\dot{x}_G^O)_N\, tan(\Phi'_N) - (\dot{y}_G^O)_N\right]\left[(\dot{x}_G^O)_N + (\dot{y}_G^O)_N\, tan(\Phi'_N)\right]}{\left[1+tan^2(\Phi'_N)\right]\mu_s g} \tag{6.20a}$$

$$Y_V' = \frac{\left[\left(\dot{x}_G^O\right)_N + \left(\dot{y}_G^O\right)_N \tan(\Phi'_N)\right]^2}{2\left[1+\tan^2(\Phi'_N)\right]\mu_s g} \tag{6.20b}$$

The sliding can be shown to stop at time T_s, given by, $T_S = \frac{2s'_N}{7\mu_s g}$, when the ball is at S (see Figure 6.5) or the coordinate (X_s',Y_s') in the $X'Y'$ system (where s'_N is the slip velocity of O on the table at the termination of impact, obtained at the Nth iteration of the numerical scheme). It is also possible to show, also using the expression in (6.18), that,

$$X_S' = \frac{2s'_N}{7\sin(\beta_N - \psi)\sqrt{\left[\left(\dot{x}_G^O\right)_N\right]^2 + \left[\left(\dot{y}_G^O\right)_N\right]^2}} X_V' \tag{6.21}$$

Y_S' can be calculated using equations (6.21) and (6.19).

The velocities along X' and Y' at the termination of slip (at S),

$$\dot{X}_S' = V_{X'}(0) = \left(\dot{x}_G^O\right)_N \cos\psi + \left(\dot{y}_G^O\right)_N \sin\psi \tag{6.22a}$$

$$\dot{Y}_S' = V_{Y'}(0) - \mu_s g T_S = -\left(\dot{x}_G^O\right)_N \sin\psi + \left(\dot{y}_G^O\right)_N \cos\psi - \frac{2s'_N}{7} \tag{6.22b}$$

The final velocity at the end of sliding is,

$$V_S = \left(\dot{X}_S'\right)^2 + \left(\dot{Y}_S'\right)^2 \tag{6.22c}$$

At an angle of θ_S with respect to the XY coordinates, given by

$$\theta_S = \psi + \lambda = \psi + \tan^{-1}\left(\frac{\dot{Y}_S'}{\dot{X}_S'}\right) \tag{6.22d}$$

Equations (6.22 a-d) completely define the post-slip motion of the ball, except its sidespin. The sidespin of the ball immediately after impact is also estimated using the numerical algorithm. Section 7.1.1.2 explains how to obtain the value of sidespin at the end of slipping process.

If the velocity and spin conditions, while the ball is still under the slipping process, are to be estimated (i.e. well before the ball reaches S) then the following set of equations will be useful. In Section 7.1.1.3, a similar situation arises.

Slip velocities along X and Y: $s_X = V_X - R\omega_Y$ and $s_Y = V_Y + R\omega_X$ (these are equivalent to equations (6.13b))

Slip velocities along X and Y in terms of its initial conditions:

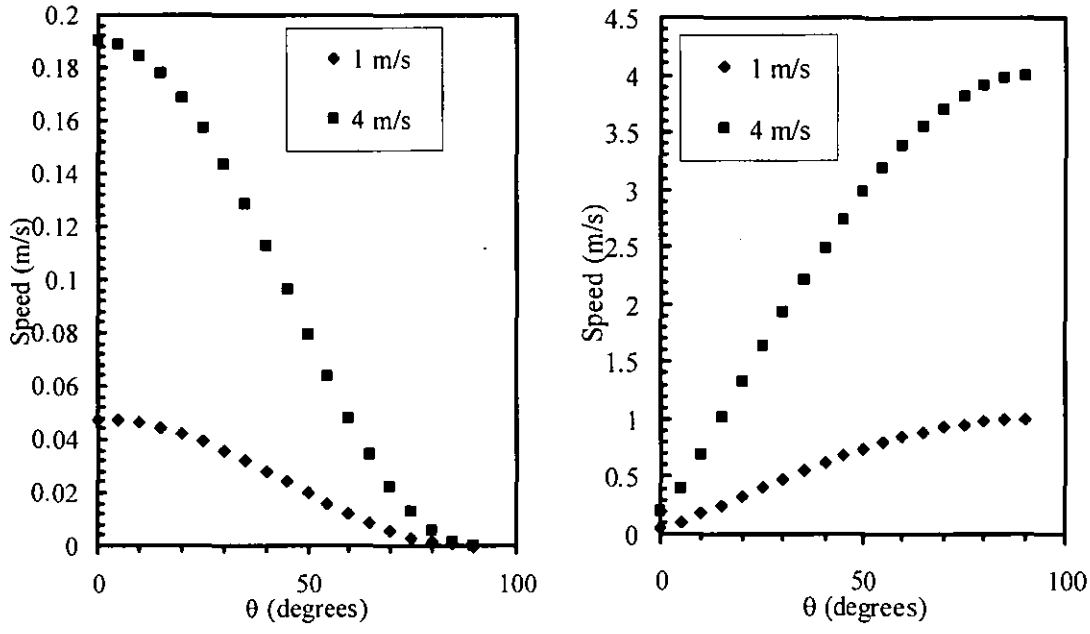
$$s_X = \left(\dot{x}_B^O\right)_N - \frac{7}{2}\mu_s g t \cos\Phi'_N \text{ and } s_Y = \left(\dot{y}_B^O\right)_N - \frac{7}{2}\mu_s g t \sin\Phi'_N$$

Time-dependant coordinates of the ball in X'Y' system: $x' = V_{X'}(0)t \;\; y' = V_{Y'}(0)t - \frac{\mu_s g}{2}t^2$

6.1.6 Motion of the Airborne Ball

In Section 6.1.2, it was explained that when $\dot{z}_A < 0$ (i.e. the relative velocity of sphere C relative to sphere O along the Z axis is in downward direction), the cue ball is not in contact with the table but is airborne instead; when $\dot{z}_A > 0$, the object ball will, instead, behave in this manner. The implication is, when $\dot{z}_A < 0$, the cue ball will have a net upward velocity (along the Z axis referring to Figure 6.2) at the end of the percussion, and the value of this velocity can be obtained from the numerical algorithm. Using the numeric value of the initial velocity of the airborne ball and its direction, the trajectory of that ball during its flight can be estimated. The following analysis of airborne trajectory is carried out for the motion of the cue ball. A similar procedure can be carried out for the object ball as well.

Figure 6.6 shows the vertical and the horizontal velocity components at the start of the cue ball flight for collision speeds of 1 m/s and 4 m/s under the conditions of pure rolling and no sidespin.



(a) (b)

Fig.6.6. Vertical (a) and horizontal (b) components of cue ball velocity at the end of impact for initial cue ball speeds of 1 and 4 m/s (it was assumed to be rolling prior to impact and no sidespin)

It can be understood from Figure 6.6 that the vertical velocity is significant when compared to the horizontal component only for head-on collisions (around θ=0). For high values of θ, the trajectory of the cue ball in the air will be more like that shown in the first drawing in Figure 6.7. The lift will be very low when compared to the horizontal range of the ball, which can be easily estimated using the theory of projectiles. The ball will land on the table with a very shallow angle to the horizontal, 'grazing' the table. As illustrated in Figure 6.7, a frictional percussion of $\mu_s P$ occurs, where P denotes the normal percussion from the table surface, which will be very small as the vertical lift itself is comparably small. As the value of μ_s is also known to be 0.21 from Chapter 5, the effects of $\mu_s P$ on the subsequent linear and angular velocities of the cue ball will be neglected. However, during the flight of the ball there will be no changes in either the horizontal velocity of the ball or in its spin. Thus, the cue ball will land on the table at a distance of $2\frac{\left(\dot{z}_G^C\right)_N}{g}\left\{\left[\left(\dot{x}_G^C\right)_N\right]^2+\left[\left(\dot{y}_G^C\right)_N\right]^2\right\}^{0.5}$, from the collision location along the direction of $tan^{-1}\left[\frac{\left(\dot{y}_G^C\right)_N}{\left(\dot{x}_G^C\right)_N}\right]$. The distances covered for the

cue ball with rolling velocities of 1, 2, 3 and 4 m/s are shown in Fig. 6.8. For a very high velocity of 4 m/s, the maximum distance obtained is 60 mm.

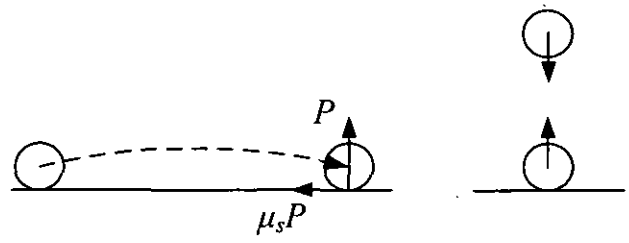


Fig.6.7. Cue ball trajectories in the air

For lower values of θ, the ball can be considered as it lifts up vertically (neglecting the horizontal component) and then falls down to the same spot where it collided with the object ball. The impulse from table will be P along the vertical and will have no effect either on the horizontal velocity of the cue ball nor on its spins about any of the three axes, as the moment of impulse created by P will be zero about all three axes. Effectively, the cue ball in its motion has a time delay whereby its dynamics are put on hold for a time of $2\frac{(\dot{z}_G^C)_N}{g}$, that is the time taken for the ball to come down again.

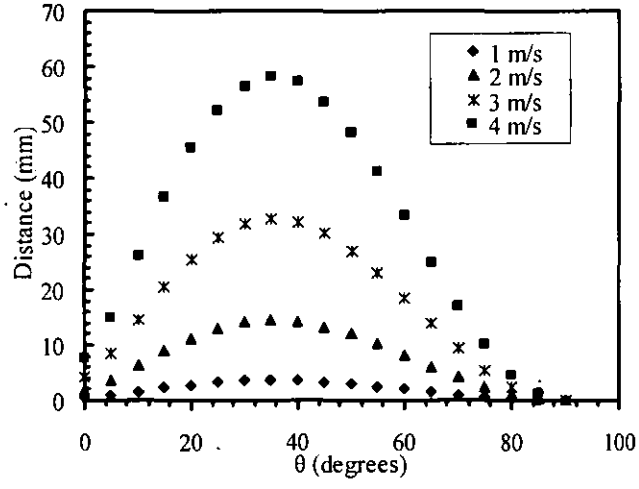
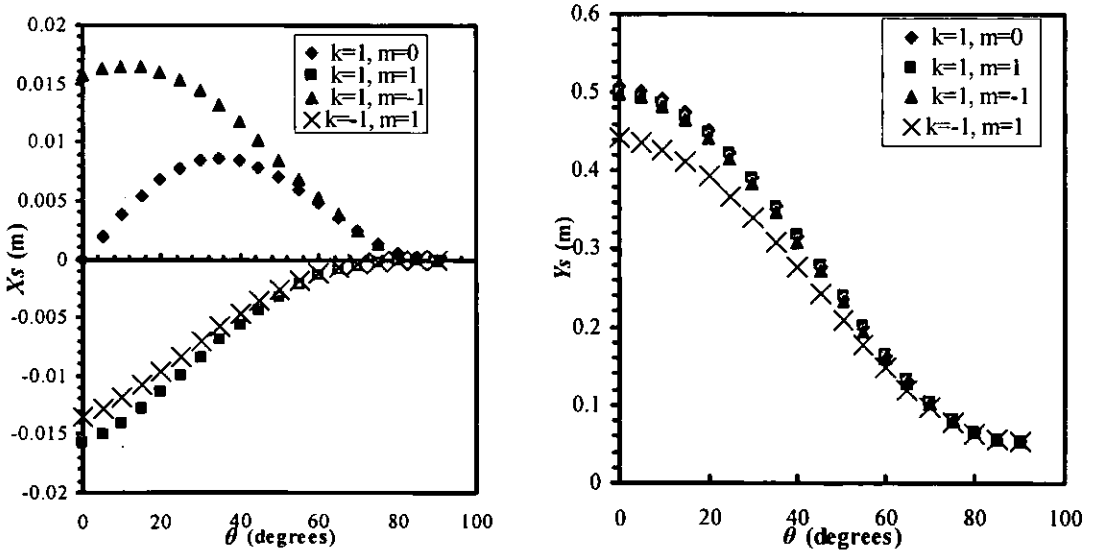


Fig.6.8. Horizontal distance covered by the cue ball during its flight

As soon as either of these types of motion, as given in Figure 6.7, is completed, the subsequent curved trajectory of the cue ball is calculated in the same manner as that for the object ball, since all its initial conditions are known at that instance.

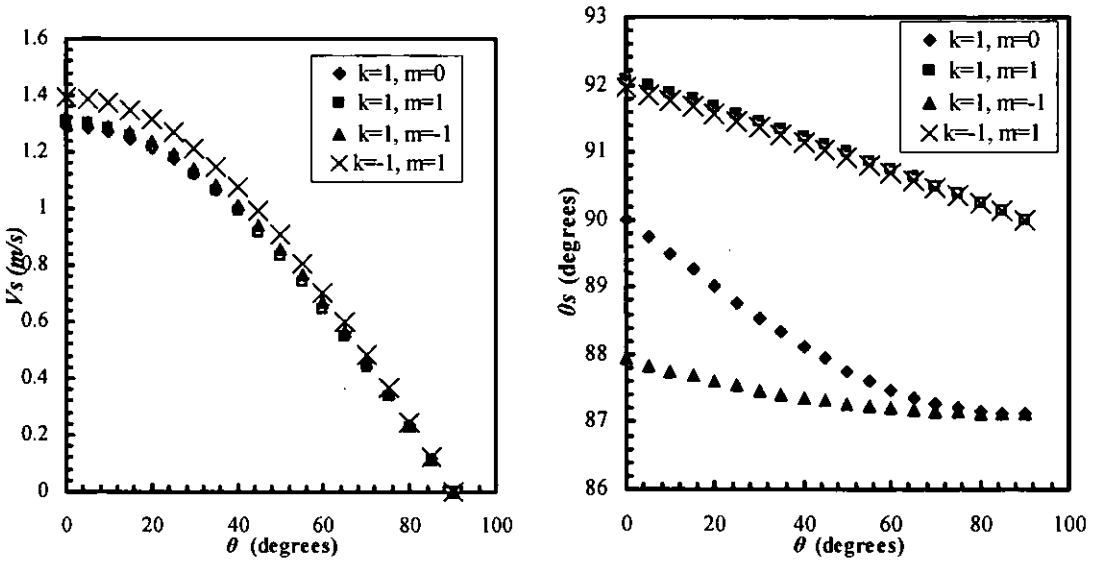
6.1.7 Results

The numerical results obtained in conjunction with equations (6.22 a-d), for the motion of cue ball and object ball, generate the plots given in Figures 6.9, 6.10, 6.11 and 6.12, for an incoming cue ball speed of 2 m/s (equations (6.22 a-d) were also used to estimate the effects due to massé). When X_s and Y_s (these are the corresponding values of X_s' and Y_s' in the XY coordinate system) were estimated, the effects of ball flight described in Section 6.1.6 were also included. Notably, when the cut angle θ changes, the XY coordinate system will also be rotating about C (see Figure 6.1). Hence the plots do not have the initial cue ball movement direction as their reference. In Figure 6.5, the values of X_s and $\psi+\lambda$ (i.e. θ_s) for the object ball are the only parameters that affect the predictions of Wallace and Schroeder **[1988]**, as described in Sections 2.3.3 and 5.1.3 respectively. Therefore, these two parameters influence the ball-potting accuracy of the robotic system. Their variations, with the ball cutting angle θ and the type of spin the cue ball has, are illustrated in Figures 6.9(a) and 6.10(b) for a cue ball speed of 2 m/s. The maximum deviation from the prediction of the 90° rule for the object ball motion occurs when the pre-collision cue ball has sidespin, as seen from Figures 6.9(a) and 6.10(b). When calculating the Y_s value for the object ball, its centre C's shift of $2R$ along the Y-axis from the origin C was also incorporated (see Figure 6.1).



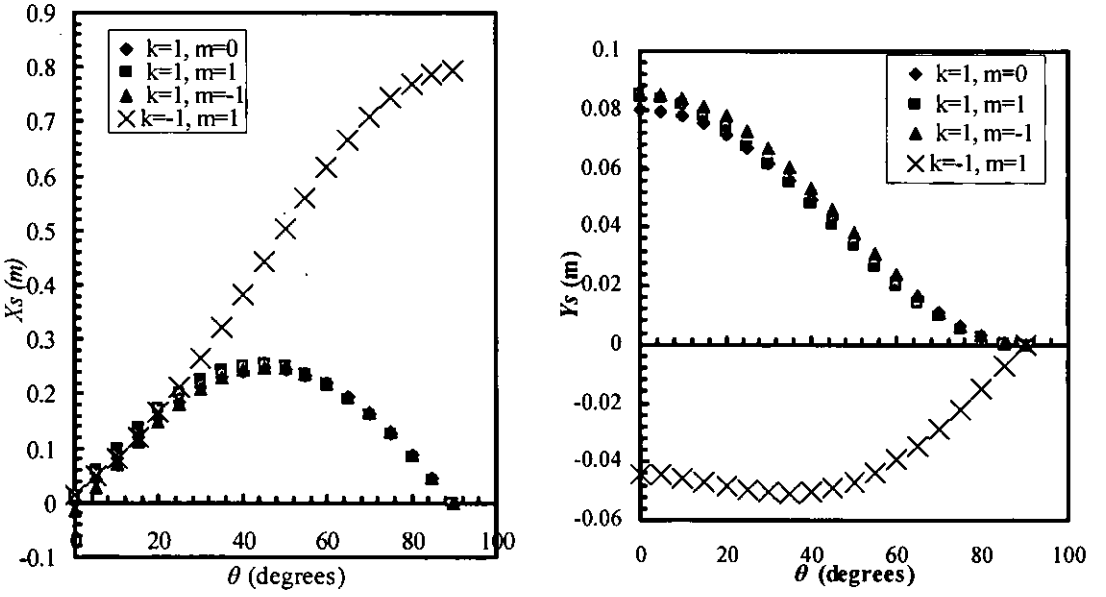
(a) (b)

Fig.6.9. For the object ball, the distances at which the slip stops, X_s and Y_s, against θ for four different shots with V_0=2 m/s ($\omega^T_0 = kV_0/R$, $\omega^S_0 = mV_0/R$)



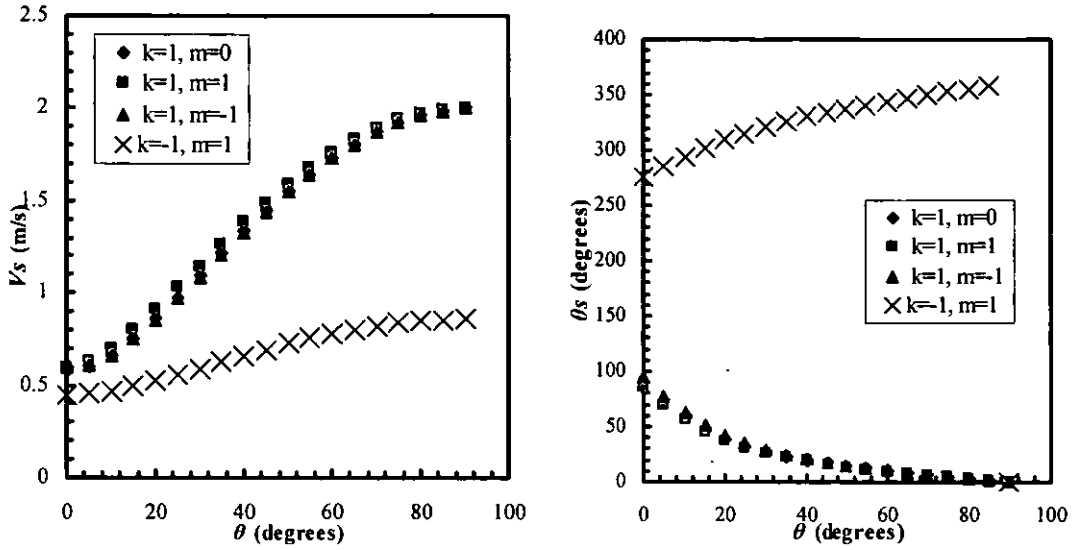
(a) (b)

Fig.6.10. For the object ball, the exit velocity and its direction after the termination of slip, V_s and θ_s, against θ for four different shots with V_0=2 m/s ($\omega^T{}_0 = kV_0/R$, $\omega^S{}_0 = mV_0/R$)



(a) (b)

Fig.6.11. For the cue ball, the distances at which the slip stops, X_s and Y_s, against θ for four different shots with V_0=2 m/s ($\omega^T{}_0 = kV_0/R$, $\omega^S{}_0 = mV_0/R$)



(a) (b)

Fig. 6.12. For the cue ball, the exit velocity and its direction after the termination of slip, V_s and θ_s, against θ for four different shots with V_0=2 m/s ($\omega^T_0 = kV_0/R$, $\omega^S_0 = mV_0/R$)

Notably, whenever the cue ball has backspin (given by $\omega^T_0<0$) before the collision, its post collision motion is directed back towards the side from which it approached the object ball. This effect is clearly shown in Figures 6.11(b) and 6.12(b), as $Y_s<0$ and $180°<\theta+\theta_s<360°$ (the addition of θ and θ_s only signifies the reversal in the direction of motion of the cue ball, also see Figures 6.1 and 6.5). This condition is satisfied for the case of $\theta<35°$ for the shot with backspin (k=-1) shown in Figure 6.12(b). This phenomenon is also depicted in Figure 2.2, where a power shot with backspin is shown to draw back and collide into a cushion resulting in the cue ball finally ending up on the side of the table from which it started.

Figure 6.13 shows the time taken for the cue ball slip on the table to terminate for different θ's. However, the airborne time of the cue ball for the three cases with $\omega^T_0 = V_0/R$ (k=1) is not included in these plots and only the time that the cue ball was in contact with the table is given in Figure 6.13. This plot, as explained in Section 7.1.1.3, is useful in estimating the sidespin of the ball after the termination of slip. When $\omega^T_0 = V_0/R$ it can be seen from Figure 6.13 that the time taken for slip remains almost the same irrespective of sidespin on the cue ball.

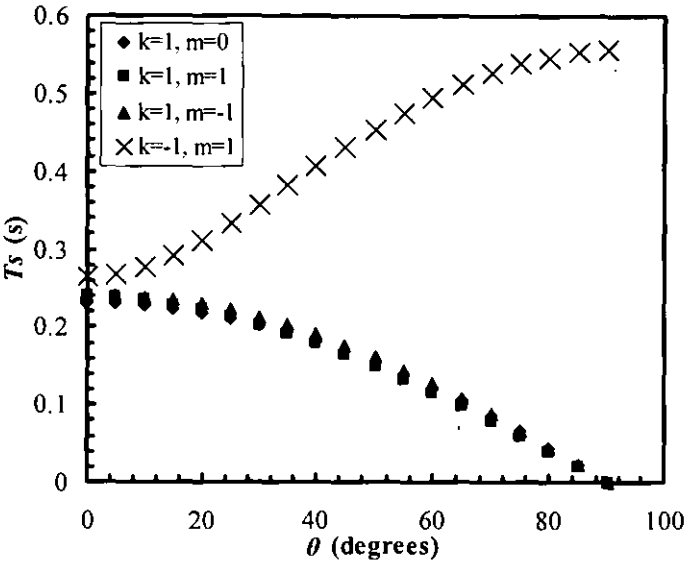


Fig. 6.13 The time taken for the cue ball's slip to terminate on the table, T_s, versus θ for four different shots with V_0=2 m/s ($\omega^T_0 = kV_0/R$, $\omega^S_0 = mV_0/R$)

6.1.7.1 Process of Slip

The numerical algorithm explained in Section 6.1.4.2 also computes the slip speeds and the slip directions between the balls and those between the ball and the table, for each increment of ΔP_I. When the slip speeds and their directions are plotted against the respective impulse values at the ball-ball interface, the shapes of the plots typically resemble the one given in Figure 6.14, which is obtained for the conditions of V_0=3 m/s, θ=45°, $\omega^S_0 = V_0/R$ and $\omega^T_0 = V_0/R$. In Figure 6.14, Φ' remains a constant while the maximum change in the value of Φ is 0.6 % of its starting value. Hence, for all practical purposes Φ and Φ' can be considered constants. This suggests that the analysis performed by Domenech **[2008]** is complete and accurate, as far as billiard ball collisions are considered. However, Domenech has not explicitly set out the justifications for this assumption of constant slip directions, which is only true for very small values of the friction coefficient between the balls (i.e. μ_{bb}), hence usable for billiard ball collisions where μ_{bb} is 0.05. For example when μ_{bb} is increased to a large value, say 0.4, the value of Φ changes drastically with P_I. Figure 6.15 shows that, for the same collisions as given in Figure 6.14, the value of Φ can change between zero and 2π radians. For such a case, the assumption made by Domenech **[2008]** will not be valid.

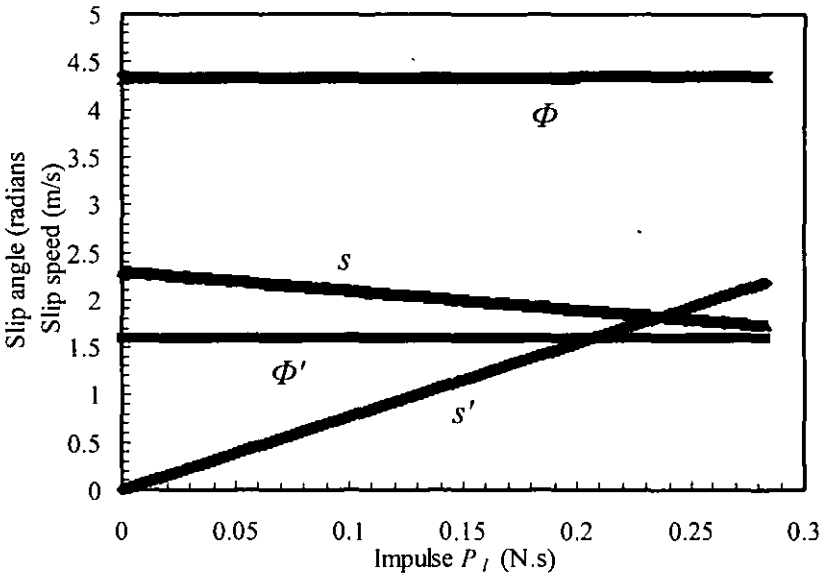


Fig. 6.14 With $\mu_{bb} = 0.05$, slip-impulse curves for V_0=3 m/s, θ=45°, $\omega^S{}_0 = V_0/R$ and $\omega^T{}_0 = V_0/R$

(s and Φ are for the slip between the balls, and s' and Φ' are for the slip at the table-object ball contact point)

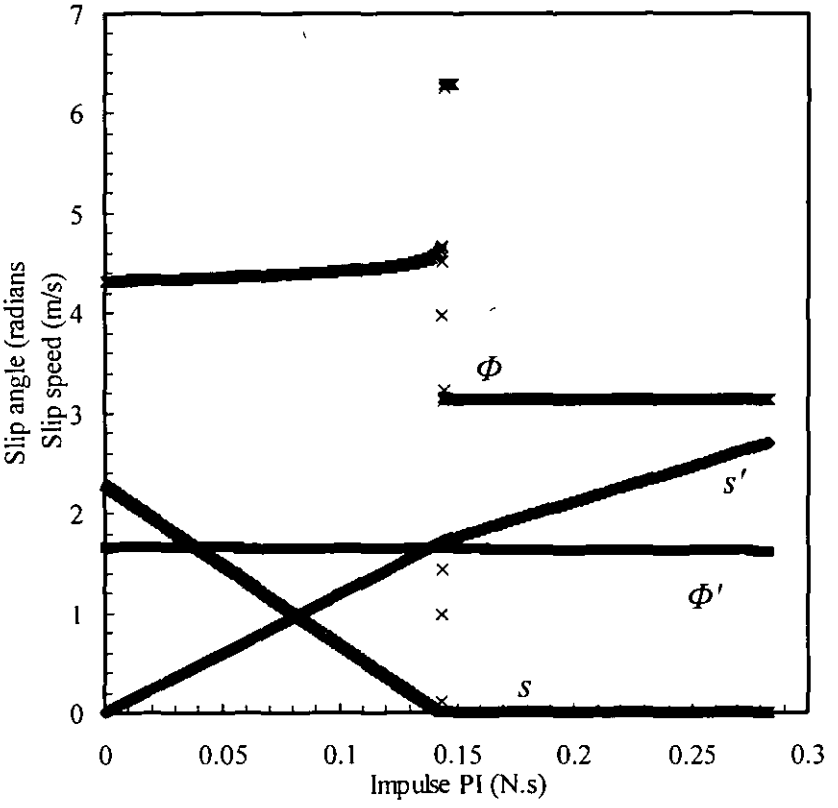


Fig. 6.15 With $\mu_{bb} = 0.4$, the slip-impulse curves for V_0=3 m/s, θ=45°, $\omega^S{}_0 = V_0/R$ and $\omega^T{}_0 = V_0/R$

6.2 Ball Collisions with the Cushion

A player often uses cushion impacts to achieve planned trajectories. Cushion impacts give a great deal of variation to the game. When combined with the effects of ball spin, the ball-cushion impacts change the ball trajectories dramatically, and give the player a greater flexibility in his game strategy (see Figure 2.2).

So far bounces of the ball off the cushion are analysed incorporating the coefficient of restitution between the ball and cushion as the only influencing parameter. In this analysis, the ball velocity normal to the cushion is considered as the sole variable. According to the current level of theory, referring to Figure 6.14, once the ball bounces off the cushion, it will have a velocity of $e_e V_0 \sin\alpha$ normal to the cushion, and a velocity $V_0 \cos\alpha$ along the cushion respectively, where e_e is the coefficient of restitution between the ball and cushion. This simple analysis does not consider the effects of ball spin and the effect of friction during the impact, and is purely treated as a two dimensional phenomenon (the plane of analysis is as given in Figure 6.16).

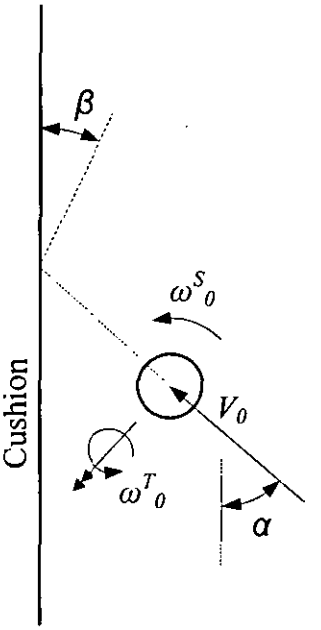


Fig. 6.16. Billiard ball prior to collision with a cushion

Spin on the ball, both sidespin ω^S_0 and topspin ω^T_0 as shown in Figure 6.16, affects both the rebound speed and rebound angle, β, of the ball. The latter two quantities are

vital to estimate the trajectory of the ball. Even though Marlow **[1994]** has tried to address these issues, the way the analysis was performed involved unnecessary parameters like the impact time between the cushion and ball for which the values were not known. Other assumptions, such as taking sliding directions as constant, do not seem correct (it is shown later that these actually change during the time impact). Marlow's analysis is not, most importantly, complete so that any comparison of the results is impossible.

The billiards cushion is made out of pure gum rubber that has good rebound properties. The cross section of a typical billiard cushion is shown in Figure 6.15. Usually, a slope is provided in the cushion such that its contact point on the ball is always above the horizontal great circle of the ball, in order to prevent the ball from leaping up in the air after impact. Here, the cushion is assumed not to change its shape during the impact with the ball, i.e. it is treated as a rigid part. This assumption may not be valid at higher ball speeds, as the normal ball velocity at I (see Figure 6.17), will try to lift up the tip of the cushion. Also, the ball and cushion are assumed to have a point contact, which again may not be true at larger ball speeds, as the ball will start to 'sink' more into the rubber cushion.

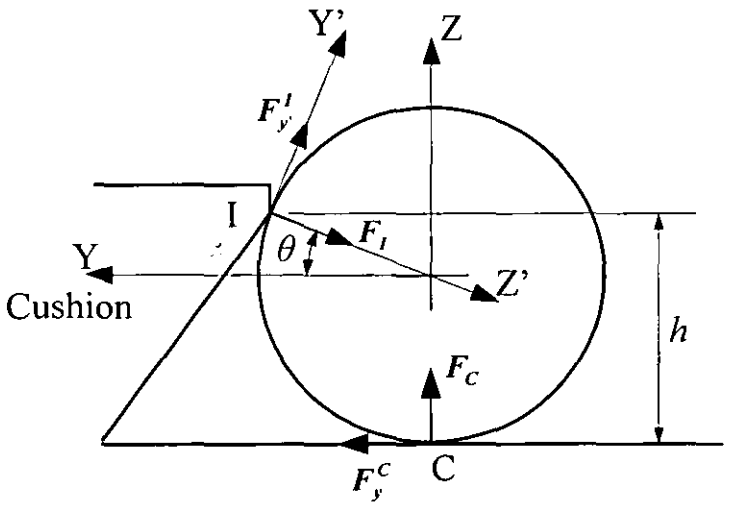


Fig. 6.17. The forces acting on the ball during the collision – a side view along the cushion at the table level

In Figure 6.17, the height of the contact point at the rail is h. In both snooker and pool h=7R/5, where R is ball radius. At the contact point with the cushion, the common normal line Z' makes an angle of θ with the Y-axis, thus, $\sin\theta = \frac{2}{5}$.

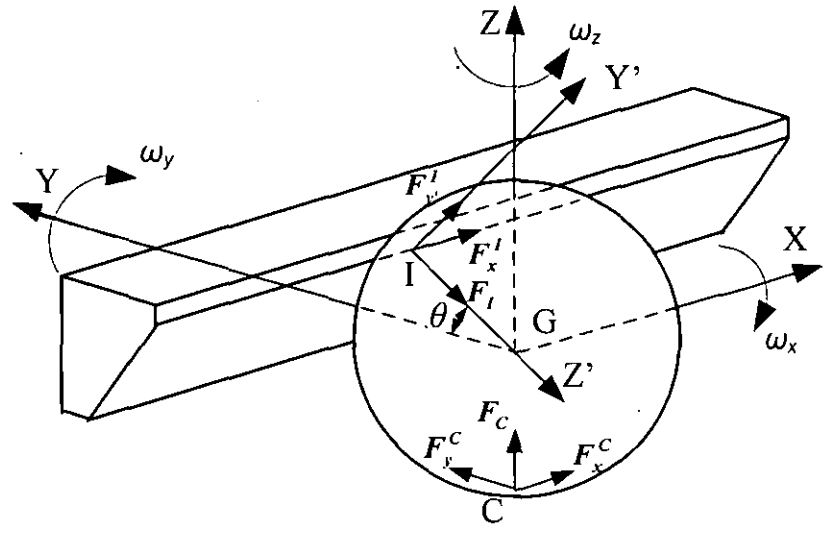


Fig. 6.18. The forces on the ball during impact (a part of the cushion is shown)

The analysis here follows the same steps taken when analysing the ball-ball collisions. Hence, only important steps are shown and wherever similarities are found, the corresponding sections under Section 6.1 are referred.

6.2.1 General Equations of Motion

Referring to Figure 6.18, the impulse-momentum relationships

$$\Delta P_x^I + \Delta P_x^C = M\Delta\dot{x}_G \tag{6.23a}$$

$$-\Delta P_I \cos\theta - \Delta P_{y'}^I \sin\theta + \Delta P_y^C = M\Delta\dot{y}_G \tag{6.23b}$$

$$-\Delta P_I \sin\theta + \Delta P_{y'}^I \cos\theta + \Delta P_C = M\Delta\dot{z}_G \tag{6.23c}$$

In equation (6.23c) it should also be noted that due to the physical sloped shape of the cushion, the vertical motion of the ball is constrained. Hence, $\Delta \dot{z}_G = 0$. Equation (6.23c) is rearranged as,

$$\Delta P_C = \Delta P_I \sin\theta - \Delta P_{y'}^I \cos\theta \tag{6.23d}$$

Similarly, for the rotational motion of the ball about the X-axis, the following equation can be derived, with angular velocity being denoted by ω,

$$\left(\Delta P_{y'}^I + \Delta P_y^C\right)R = I\Delta\omega_x$$

where, the moment of inertia of the ball, $I = \frac{2MR^2}{5}$. The above equation can be written as:

$$\Delta P_{y'}^I + \Delta P_y^C = \frac{2MR}{5}\Delta\omega_x \tag{6.24a}$$

Similarly about the Y and Z axes:

$$\Delta P_x^I \sin\theta - \Delta P_x^C = \frac{2MR}{5}\Delta\omega_y \tag{6.24b}$$

$$-\Delta P_x^I \cos\theta = \frac{2MR}{5}\Delta\omega_z \tag{6.24c}$$

6.2.2 Impact Dynamics at Locations I and C

At the contact point with the cushion, I, the slip will take place on the XY' plane (i.e. the tangential plane), also noting that the axis Y' is in the YZ plane. Let the slip speed of the ball at I, be $s(t)$ at an angle $\Phi(t)$ from the X-axis.

The slipping velocities along the X and Y' axes are given by, respectively,

$$\dot{x}_I = s(P_I)\cos(\Phi(P_I)) \tag{6.25a}$$

$$\dot{y}'_I = s(P_I)\sin(\Phi(P_I)) \tag{6.25b}$$

However, $\dot{y}'_I$ can also be written as,

$$\dot{y}'_I = -\dot{y}_I \sin\theta + \dot{z}_I \cos\theta \tag{6.26}$$

Using the Amontons-Coulomb law of friction, for $s > 0$, also noting that the friction forces/impulses are opposite to the direction of sliding, the friction impulses along X and Y' are

$$\Delta P_x^I = -\mu_w \cos(\Phi(P_I))\Delta P_I \quad (6.27a)$$

$$\Delta P_{y'}^I = -\mu_w \sin(\Phi(P_I))\Delta P_I \quad (6.27b)$$

where, μ_w is the coefficient of friction between the ball and the cushion.

From (6.23d) and (6.27b), the normal reaction from the table surface to the ball is given by,

$$\Delta P_C = (\sin\theta + \mu_w \sin(\Phi(P_I))\cos\theta)\Delta P_I \quad (6.28)$$

Using the earlier argument, for the impact at C, the instantaneous impulse value P_C should be chosen as the independent variable. But equation (6.28) shows that the value of P_C directly depends on the value of P_I. Hence, also for the impact at C, P_I is considered as the independent variable. This makes it possible to have P_I as the independent variable for all the impulse forces involved in this analysis.

For the impact at C, the slip takes place on the XY plane. Let s' be the slip speed, and Φ' be the direction of slip measured from the X-axis. Now, the components along the X and Y directions are,

$$\dot{x}_C = s'(P_I)\cos(\Phi'(P_I)) \quad (6.29a)$$

$$\dot{y}_C = s'(P_I)\sin(\Phi'(P_I)) \quad (6.29b)$$

Hereafter let us omit the independent variable P_I from all equations for the sake of simplicity. When $s' > 0$, the expressions for the impulse forces along X and Y directions, also using equation (6.28), are

$$\Delta P_x^C = -\mu_s \cos\Phi' \Delta P_C$$

$$= -\mu_s \cos\Phi'(\sin\theta + \mu_w \sin\Phi\cos\theta)\Delta P_I \quad (6.30a)$$

$$\Delta P_y^C = -\mu_s \sin\Phi' \Delta P_C$$

$$= -\mu_s \sin\Phi'(\sin\theta + \mu_w \sin\Phi\cos\theta)\Delta P_I \quad (6.30b)$$

Where μ_s is the coefficient of friction between the ball and table surface.

6.2.2.1 Conditions for Rolling

1) When s=0, the ball will be rolling on the cushion at , neglecting the effects of stick,

$\Delta P_x^I = \Delta P_{y'}^I = 0$ and from (3c), $\Delta P_C = 0$. Hence, $\Delta P_x^C = \Delta P_y^C = 0$ (6.37a)

2) When s '=0, the ball will roll on the table surface, and neglecting stick,

$$\Delta P_x^C = \Delta P_y^C = 0 \tag{6.37b}$$

In Section 5.1.1.2, that the average value of μ_s was found to be 0.21.

6.2.2.2 Coefficient of Restitution

Work done at I, along the axis Z', is

$$\Delta W_{Z'_I} = \int_t^{t+\Delta t} F_I . \dot{z}'_I \, dt = \int_{P_I}^{P_I+\Delta P_I} \dot{z}'_I \, dP_I$$

Its numerical form is,

$$\left(W_{Z'_I}\right)_{n+1} - \left(W_{Z'_I}\right)_n = \Delta P_I \frac{\left[\left(\dot{z}'_I\right)_{n+1} + \left(\dot{z}'_I\right)_n\right]}{2} \tag{6.38a}$$

The coefficient of restitution is given by,

$$e_e{}^2 = \frac{-\int_{P_I^c}^{P_I^f} \dot{z}'_I \, dP_I}{\int_0^{P_I^c} \dot{z}'_I \, dP_I}$$

Rearranging the equation,

$$W_{Z'_I}\left(P_I^f\right) = \left(1 - e_e{}^2\right) W_{Z'_I}\left(P_I^c\right) \tag{6.38b}$$

The termination of compression occurs when the normal component of relative velocity becomes zero, i.e.,

$$\dot{z}'_I\left(P_I^c\right) = 0 \tag{6.38c}$$

6.2.3 Velocity Relationships

The velocity of any point on a sphere's surface can be expressed in vector notation as

$$\vec{V} = \vec{V}_G + \vec{\omega} \Lambda \vec{R}$$

The slip velocities at I are,

$$\dot{x}_I = \dot{x}_G + \omega_y R \sin\theta - \omega_z R \cos\theta \tag{6.39a}$$

$$\dot{y}'_I = -\dot{y}_G \sin\theta + \dot{z}_G \cos\theta + \omega_x R \tag{6.39b}$$

and at C,

$$\dot{x}_C = \dot{x}_G - \omega_y R \tag{6.40a}$$

$$\dot{y}_C = \dot{y}_G + \omega_x R \tag{6.40b}$$

6.2.4 Solution to the Ball Velocity

As in Section 6.1.4, the increments in centroid velocities of the ball can be calculated in terms of the instantaneous values of centroid velocities. For example, the expression for $\Delta\dot{x}_G$ can be written as:

$$(\dot{x}_G)_{n+1} - (\dot{x}_G)_n = -\frac{1}{M}\{\mu_w \cos(\Phi)_n + \mu_s \cos(\Phi')_n [\sin\theta + \mu_w \sin(\Phi)_n \cos\theta]\}\Delta P_l \tag{6.41a}$$

where, $\tan(\Phi)_n = \dfrac{-(\dot{y}_G)_n \sin\theta + (\dot{z}_G)_n \cos\theta + (\omega_x)_n R}{(\dot{x}_G)_n + (\omega_y)_n R \sin\theta - (\omega_z)_n R \cos\theta}$

and, $\tan(\Phi')_n = \dfrac{\dot{y}_G + \omega_x R}{\dot{x}_G - \omega_y R}$

For the other 5 components of the centroid velocity, similar expressions to (6.41a) can be written.

6.2.4.1 Initial Conditions

$(\dot{x}_G)_1 = V_0 \cos\alpha$, $(\dot{y}_G)_1 = V_0 \sin\alpha$, $(\dot{z}_G)_1 = 0$, $(\omega_x)_1 = -\omega_0^T \sin\alpha$, $(\omega_y)_1 = \omega_0^T \cos\alpha$, and $(\omega_z)_1 = \omega_0^S$.

$$s(0) = \left|\sqrt{\left[V_0 \cos\alpha + R\left(\omega_0^T \cos\alpha \sin\theta - \omega_0^S \cos\theta\right)\right]^2 + \left[-V_0 \sin\alpha \sin\theta - R\omega_0^T \sin\alpha\right]^2}\right|$$

$$s'(0) = \left|V_0 - R\omega_0^T\right|$$

The numerical algorithm follows the same steps as those for a ball-ball collision (see Section 6.1.4.2).

6.2.5 Results

6.2.5.1 Estimating e_e and μ_w

The experimental plot in Figure 5.7 was used to calculate the values of the coefficient of restitution e_e, and the sliding friction coefficient between the ball and cushion μ_w. It is known that $0<e_e\leq 1$. The results exhibited in Figure 5.7 are obtained under the conditions of $\alpha = 90°$, $\omega_0^S = 0$ and $\omega_0^T = \frac{V_0}{R}$. Under these conditions, for each of the experimentally-obtained incident speed values (i.e. V_0) in the speed range of $V_0<1.5$ m/s, in Figure 5.7, the numerical algorithm was run for values of e_e and μ_w starting from 0 and increasing to 1 in increments of 0.01, and the rebound speed $\dot{y}_G\left(P_I^f\right)$ was obtained. Higher speeds were not considered, as the assumption of a rigid cushion may not then be applicable. The values of e_e and μ_w that minimise the RMS value of all errors between the experimental and numerically-predicted rebound speeds should be the actual value for the coefficient of restitution between the cushion and ball. Calculations showed that e_e=0.98 and μ_w=0.14.

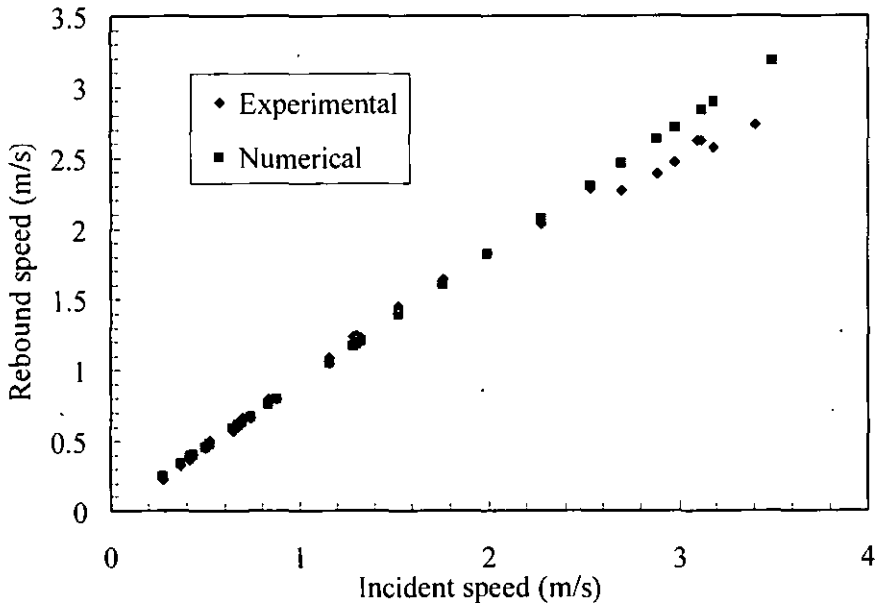


Fig. 6.19. Rebound speed versus incident velocity for e_e=0.98 and μ_w=0.14

Numerically-obtained rebound speed values for e_e=0.98 are plotted in Figure 6.19 together with the experimentally-obtained values. As seen in Figure 6.19, the

numerically-obtained values of incident speed deviate from the experimentally-obtained values for speeds V_0>2.5 m/s. This is, quite possibly, the velocity limit at which the rigid body assumption for the cushion would be valid. V_0=2.5 m/s is a considerably high speed as far as snooker is concerned. For oblique shots, only the ones for which the normal component of the incident velocity of less than 2.5m/s, would be analysed using the numerical algorithm.

6.2.5.2 Ball trajectories

The outputs from the numerical algorithm show that, in most cases, the ball has some spin about its velocity axis at the termination of impact, causing it to curve. Under such conditions, the equations derived in Section 6.1.5 can be used to determine the parameters of the curved path and the exit conditions of the ball. The results obtained for a ball speed of 1m/s and 5 different spin conditions are given in Figures 6.20 and 6.21. X_S and Y_S are measured with respect to the coordinate system OXY as shown in Figure 6.18.

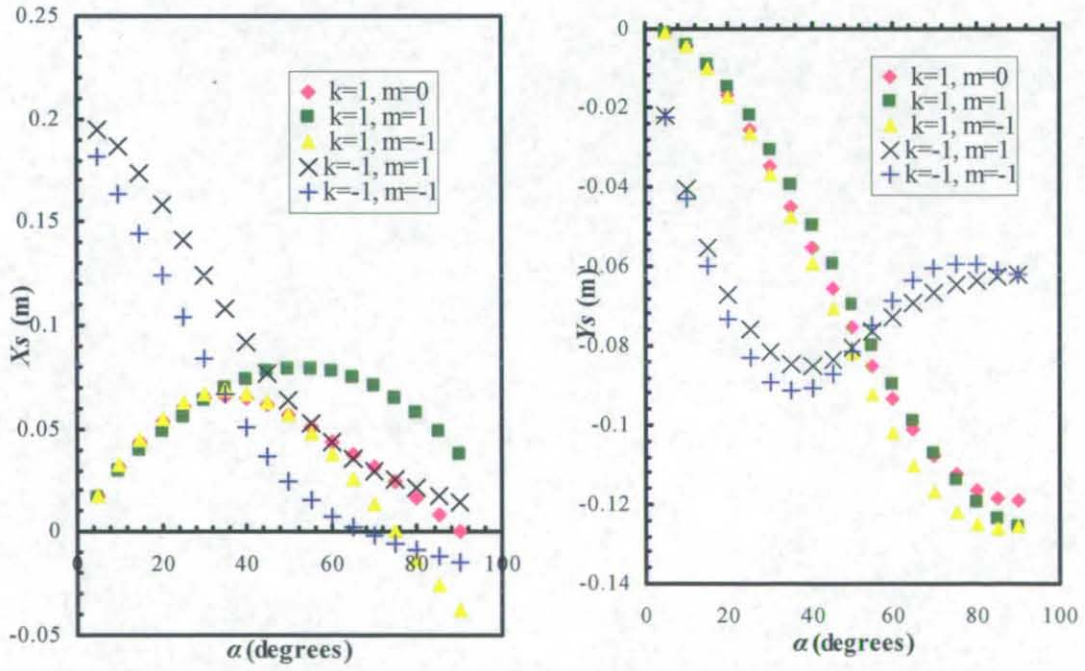


Fig.6.20. Distances at which the slip stops, X_s and Y_s, against α for four different shots with V_0=1 m/s ($\omega^T_0 = kV_0/R$, $\omega^S_0 = mV_0/R$)

For the cases where ω^S_0=-V_0/R (i.e. m=-1 in the plots), from Figures 6.20(a) and 6.21(b), it can be seen that for α>70° the value of X_s becomes negative and the direction of its exit velocity, θ_S, is below 270°. The implication is that, with reference to Figure 6.22, the ball bounces back to the same side from which it approached the cushion. This effect has been described by Walker **[1983]** for billiards, and by Cross **[2005]** in a general context for the bounce of a ball. Cross **[2005]** also presents some experimental values for a tennis ball bouncing on a rough surface.

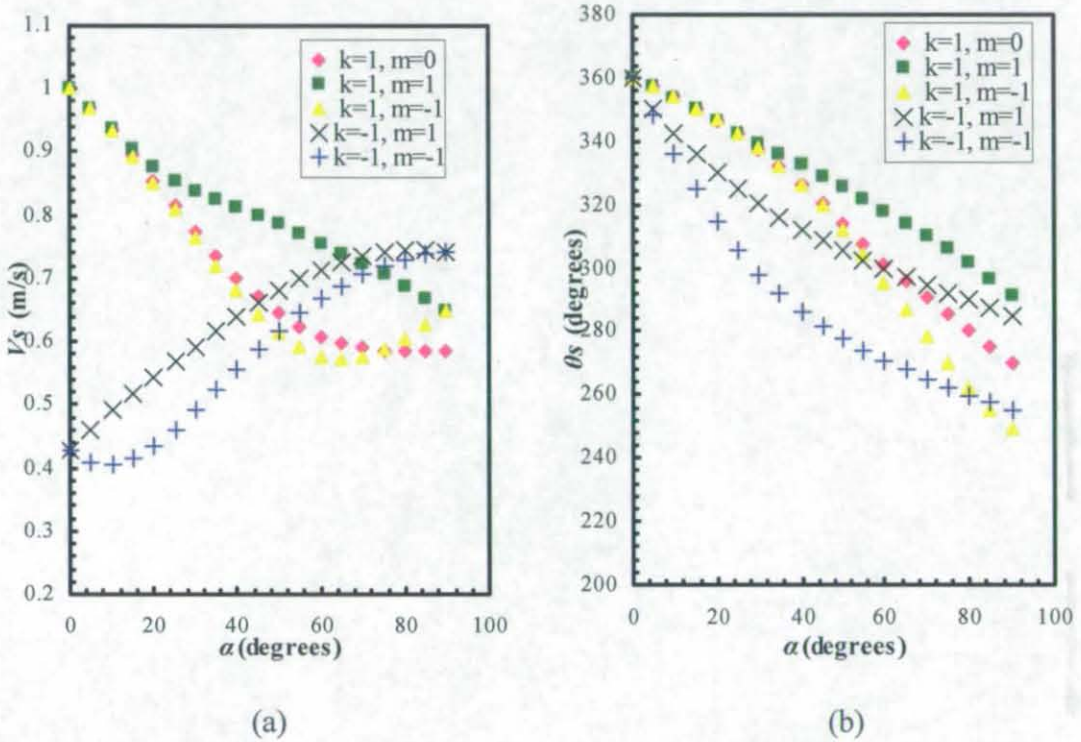


Fig.6.21. Exit velocity and its direction at the termination of slip, V_s and θ_s, against α for five different shots with V_0=1 m/s ($\omega^T_0 = kV_0/R$, $\omega^S_0 = mV_0/R$)

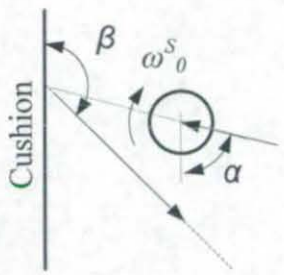


Fig. 6.22. Ball bouncing back to the same side under left spin conditions

6.2.5.3 A note on the process of slip

As stated at the beginning of Section 6.2, the slip directions have been assumed to be unchanged from the values at the onset of impulse by Marlow **[1994]**. The numerical algorithm was also programmed to calculate the directions of slip and slip speeds for every increment of ΔP_I using equation sets (39) and (40). A plot against the instantaneous impulse value is shown in Figure 6.23. The change in slip direction as indicated by the plot suggests that the assumption of uni-directional slip is wrong.

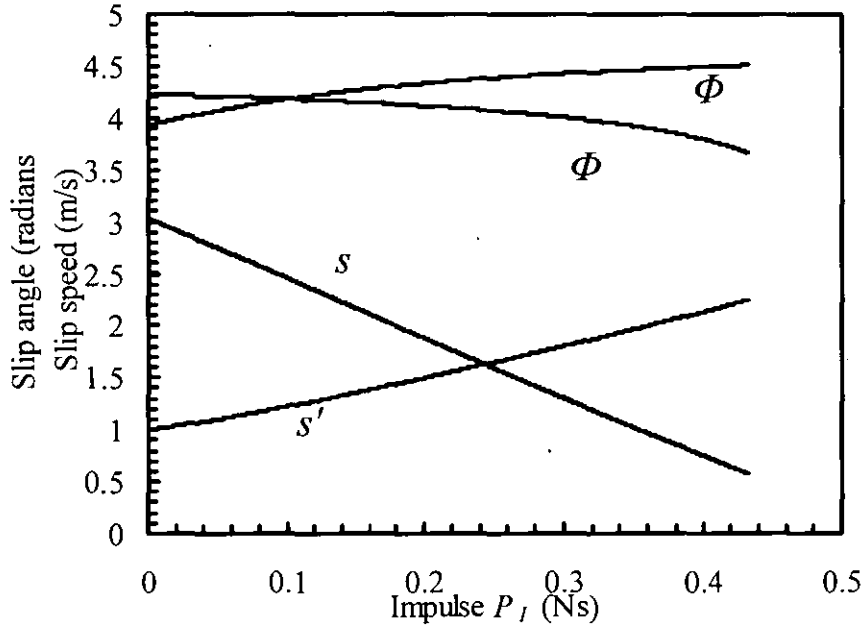


Fig. 6.23. Slip-impulse curves for V_0=2 m/s, α=45°, $\omega^S{}_0 = 2V_0/R$ and $\omega^T{}_0 = 1.5V_0/R$ (s and Φ are for the slip at the cushion, and s' and Φ' are for the slip at the table)

Summary

Using the principles of impact mechanics, the ball-ball collision was analysed. After obtaining the differential equations describing balls dynamics during the period of impact, they are solved numerically. When combined with two experimental plots from Chapter 5, the numerical solutions provide the values of the friction present between the balls and their coefficient of restitution. Numerical algorithms provide post-impact velocities and the spins of the balls for a given collision condition consisting of different ball speeds, directions and ball spins. For the first time, the object ball is shown with the application of massé-type spin, which results in a curved trajectory. A description of the massé shot trajectory is presented, suitably adapted from an earlier work on the curving effects of bowling balls. The second part of the chapter presents the same set of analyses for a ball-cushion collision. For both cases, the slip curves during impact are provided and by using these curves some assumptions adopted by earlier researchers are either validated or disproved.

Chapter 7

MANIPULATION PROBLEM: DEFINITION, SOLUTIONS AND RESULTS

This chapter focuses on the main question addressed by this thesis: how to position the balls at desired locations on the table. First of all, the manipulation problem is defined within the context of the background information from the previous chapters. The performance of the robotic system is evaluated. Also, an empirical model, using neural networks, for the cueing dynamics is obtained. Section 7.2 outlines the genetic algorithm-based optimisation procedure that is suggested as the solution for the manipulation problem. The last part provides the relevant experimental results.

7.1 Manipulation problem definition and background information

The artificial intelligence part of the system always makes decisions regarding which object ball has to be played next, the pocket in which the object ball must to be potted and where to leave the cue ball in order to make the next shot according to the overall game plan (this is discussed in Section 2.1). Thus, for a given initial cue ball location, C_I, as depicted in Figure 7.1 (only a part of the table is shown there) the decision to play the ball O_I into the pocket P_I and then to leave the cue ball in or very close to the desired ball location C_D has already been taken by the decision-making system. These results are assumed to be readily available. Now it is up to the robot dynamics planner to plan and execute the shot so that the trajectories are achieved accordingly. This planning phase is discussed in this section. The initial parameters of the ball motion are, V_0, ω^T_0, ω^S_0, θ, namely, the velocity, top and side spins imparted to the ball, and the direction along which the ball moves. These parameters of the ball motion can be varied by the controllable parameters of the robot as will be seen later in this section.

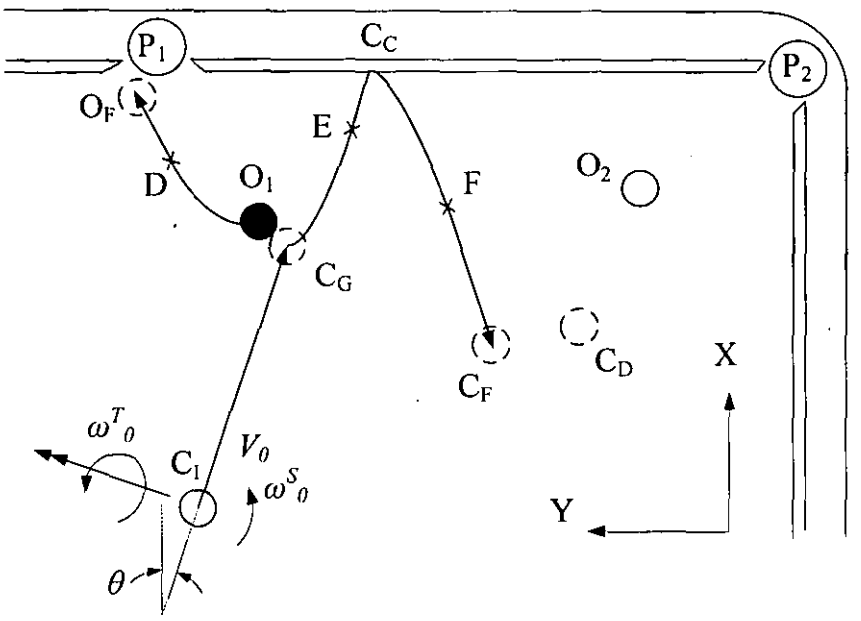


Fig. 7.1. A typical ball trajectory in snooker

The velocity of the cue determines the cue ball speed V_0. In Section 2.4.1 it was described that the hitting point on the ball and the cue velocity determines the type and amount of spin imparted to the cue ball. In summary, the cue velocity and the hitting point stand out as two important variables. The third one is the cue swivel angle θ_C (this is slightly different to θ due to cue squirt, as described in section 2.4.1.3). In the robot, the cue velocity is determined by the speed of the servomotor. The cue positioning on the ball is performed by the stepper motor unit, as described in Section 4.2.

7.1.1 Ball Trajectories on the Table

Here the important ball motion characteristics are briefly outlined.

7.1.1.1 Straight-line ball motion

The essential equations to describe the ball's straight-line movement are given in Section 2.4.2, describing its sliding and rolling motion. Parameters needed for the equations are estimated in Section 5.1.1.

7.1.1.2 Estimating sidespin of the ball

Sidespin does not affect the free motion of the ball on the table, and its value is not influenced by the linear speed of the ball (the assumption of decoupled motion). However, sidespin plays a vital part in determining the dynamics of impacts, as seen in Sections 6.1 and 6.2. This underlines the need to estimate the amount of sidespin on the ball at any time during its motion, as the ball on its path can encounter another ball or a cushion at any instant. The following formula describes the instantaneous value of sidespin at time t for its initial value of ω_0^T,

$$\omega^T(t) = \omega_0^T - \dot{\omega}_r t, \qquad \text{for } \omega_0^T > 0 \text{ and } t \le \left|\frac{\omega_0^T}{\dot{\omega}_r}\right| \tag{45a}$$

$$\omega^T(t) = \omega_0^T + \dot{\omega}_r t, \qquad \text{for } \omega_0^T < 0 \text{ and } t \le \left|\frac{\omega_0^T}{\dot{\omega}_r}\right| \tag{45b}$$

$\dot{\omega}_r$ is the resistance of the table to the sidespin of the ball, which is measured as 22 rad/s^2 in Section 5.3. It should also be noted that clockwise rotation of the ball (i.e. right-spin), as seen when looking down on the table, is taken as positive. Once $\omega^T(i)$ becomes zero, it stays at that value unless the ball collides with a ball or a cushion where it attains a new value. ω_0^T is the value of sidespin immediately after cueing or the one immediately after an impact, in which case it is estimated from the numerical schemes described in Sections 6.1 and 6.2.

7.1.1.2 Dynamics of the Ball Collisions

When the dynamics prior to the collisions are provided, the algorithms presented in Sections 6.1 and 6.2 form the basis for the estimation of the post-collision velocities and the ball's trajectory changes during the collisions.

7.1.1.3 Collisions under Massé Conditions

Cue ball collisions, either with another ball or with a cushion, can also occur during the curved phase of the ball motion shown by segments C_GE or O_1D (see Figure 7.1). Under such situations, the incoming conditions of the ball (its speed, direction of movement, frontal and top spins and its sidespin) have to be estimated using the equations given towards the end of Section 6.1.5.

Currently, the robot swivel angle, θ_C , has to be adjusted manually, thus it does not have any constraints imposed on it. Generally, in the presence of cue squirt, the direction along which the ball travels after the cueing, θ, will differ from θ_C by a certain amount, as discussed in Section 2.4.1.3. But θ_C directly influences θ and hence it is one of the manipulation parameters. However, for a given cue ball-object ball combination, for the former to impinge on the latter, θ can be shown to lie within specific limits.

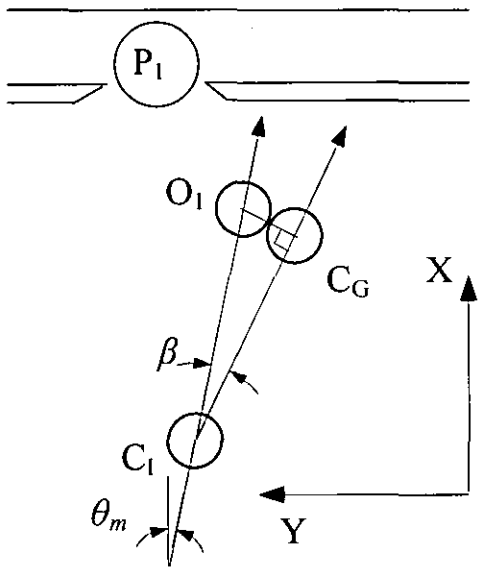


Fig. 7.2. Constraints on the cue ball's direction of movement

Let $C_I \equiv [x_C, y_C]$ and $O_I \equiv [x_O, y_O]$, be in the table coordinates XY that are defined for the imaging purposes (Figure 4.18). The inclination of O_IC_I with respect to the X-axis can be obtained from,

$$\tan\theta_m = \frac{y_O - y_C}{x_O - x_C}. \qquad (7.1)$$

The cue ball trajectory when it brushes past the object ball sets a lower limit for θ. Right at the moment the cue ball grazes against the object ball on its right side, the cue ball is at C_G on the table (see Figure 7.2). For the right-angled triangle $C_IC_GO_I$, as the radii of the cue ball and the object ball are equal,

$$\sin\beta = \frac{2R}{\sqrt{(x_O - x_C)^2 + (y_O - y_C)^2}} \tag{7.2}$$

Now, the minimum possible value for θ for collision to take place is:

$\theta_{min} = \theta_m - \beta$

Using a similar procedure, it could also be proved that:

$\theta_{max} = \theta_m + \beta$

Hence the possible values for θ are given by:

$$\theta_m - \beta \le \theta \le \theta_m + \beta \tag{7.3}$$

Thus, the robot swivel angle, θ_C, must also be kept within specific limits defined by the expression in (7.3) for the cue ball to impinge on the object ball.

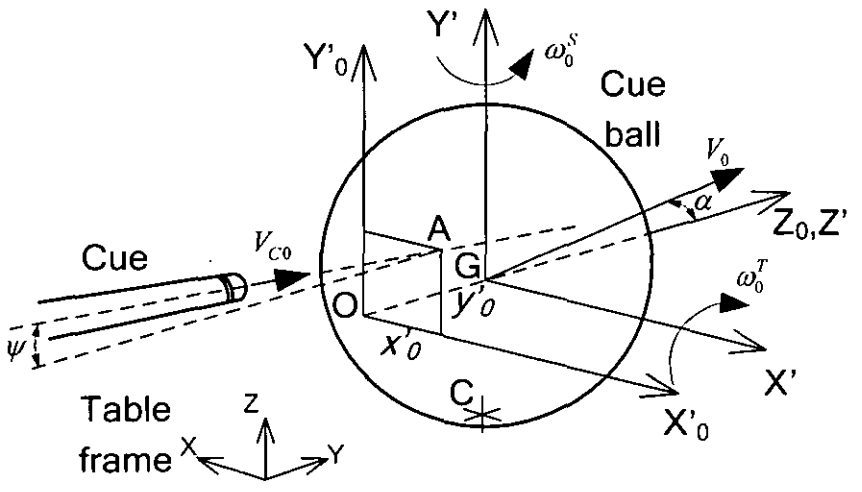


Fig. 7.3. Manipulation parameters of the robot and the initial cue ball motion

The other manipulation parameters of the robot are its linear velocity V_{C0} with which it drives the cue and the cue impact location on the ball. Referring to Figure 7.3, let O be the point on the ball where the cue should impinge on the ball to obtain the stun condition for a given position of the robot frame (essentially the stepper platform location). Hence, O will lie on the horizontal great circle of the cue ball. Let the coordinate system $X'_0Y'_0Z'_0$ be such that O is its origin and OG lies along the Z_0 axis (G is the ball centroid). X'_0 and Y'_0 are aligned to the horizontal and the vertical respectively. $X'_0Y'_0Z'_0$ essentially represents the stepper drive's coordinate system, where X'_0 and Y'_0 lie along the drive's positioning directions. Any movement x_0 and

y_0 of the stepper will result in the driving line of the cue being shifted so as to go through point, say A, on the $X'_0Y'_0$ plane that has coordinates $[x'_0, y'_0]$ (refer to Figure 7.3). Since the horizontal inclination of the cue, ψ, is known as 6.5°, the impact point on the ball can be calculated in the following manner. Now let X'Y'Z' be a coordinate system having the ball COG as its origin, which is shifted from the $X'_0Y'_0Z'_0$ system by a distance R along the Z_0 axis.

The equation of the straight-line along which the cue moves, in the X'Y'Z' system, is,

$$x' = x'_0, \text{ and, } z' + R = -\frac{(y' - y'_0)}{\tan\psi} \tag{7.4}$$

Alciatore **[2004]** recommends that the radius of the cue-tip should ideally be the radius of a nickel (a U.S. coin). The pre-shaped cue-tip that was used here was 10mm in radius, which is very close to the size suggested by Alciatore. When the cue tip touches the ball denoting the ball radius by R and the cue-tip radius by r, the Z' coordinate of the centre of the sphere representing the cue-tip surface can shown to be the solution of the following quadratic equation:

$$(x'_0)^2 + \left[y'_0 - (z'_T + R)\tan\psi\right]^2 + (z'_T)^2 = (r + R)^2 \tag{7.5}$$

There will be two values for z'_T, and the negative root is the relevant one. Now, substituting for z' using equation (7.4), the coordinates of the centre of the cue-tip surface can be worked out, let them be $[x'_T, y'_T, z'_T]$. Let the coordinates of the contact point between the cue tip and the ball be $[x'_C, y'_C, z'_C]$. They are the solutions of the following set of equations,

$$\begin{bmatrix} x'_C \\ y'_C \\ z'_C \end{bmatrix} = \frac{R}{R + r} \begin{bmatrix} x'_T \\ y'_T \\ z'_T \end{bmatrix} \tag{7.6}$$

Therefore, the impact location, which affects the spin imparted to the ball (ω^T_0 and ω^S_0), the ball velocity (V_0), and the amount of cue squirt, denoted by α in Figure 7.3, is essentially a function of the stepper movement x'_0 and y'_0.

There are constraints on each of the robot parameters. For the robot, the following limit is set,

$$V_{C0} \leq 2.2\text{m/s} \tag{7.6}$$

V_{C0} is programmed to be as high as 2.75 m/s, but due to vibrations and the associated problems with the robot's repeatability it was not taken above 2.2 m/s. This maximum cue velocity produces ball speeds close to 3 m/s, which is high enough to do the tests over the half-table area.

According to Alciatore **[2004]**, for mis-cueing not to occur,

$$\sqrt{(x'_C)^2 + (y'_C)^2} \leq \frac{R}{2} \tag{7.7}$$

Coriolis suggested that the limit is $0.7R$ **[Nadler 2005]**. x'_0, and, y'_0, (and in turn x'_C and y'_C) are not constrained by the robot's dynamics, but are to be rounded off to the nearest millimetre.

Equations (7.3), (7.6) and (7.7) define the constraints for the control parameters of the robot, V_{C0}, x'_0, y'_0 and θ_C.

7.1.3 Cueing Dynamics of the Robot

So far, a dynamic model for the ball motion on the table has been presented (as summarised in Section 7.1.1). A complete control of the robotic components (both the servo and the stepper drive) has also been established as described in Chapter 4. Now, the final task is to establish a model for the interaction between the robot and its environment (here the ball-table dynamics). The interaction is cueing, and a model is developed for this process here. For a tapping-based positioning robot, Huang and Mason **[2000]** have used Routh's method of analysing two-dimensional impacts with friction, between the tapper and the planar part that is being positioned. In their study, an analytical solution is possible as both their tapper and the tapped object are rigid objects for which a theoretical analysis is a straightforward possibility. However, in the case of the cue and the ball colliding, the cue-tip, which is both soft and deformable, is present at the collision interface. The presence of the cue tip complicates the dynamics at the interface.

Before the cueing dynamics of the robot can be determined, the performance of the cue launcher system has to be evaluated for its consistency.

7.1.3.1 Performance of the Robot

The servomotor-based cueing element is controlled through an Iensys® microcontroller board that is programmed to send pulses at a constant rate to the servo controller. When the pulse rate is at its highest, the linear striking velocity of the cue reaches 2.75 m/s, a typical maximum cue velocity found in a normal game of snooker (some speed measurements on typical human cueing are shown in Figure 5.16). The rate at which the pulses are sent out from the microcontroller is selected by a string consisting of a 3 digit number appended with a 'p' from the PC through its serial port to the serial interface of the microcontroller. This 3-digit string, which ranges from'001 'to'200', selects the intended pulse rate. String '001'corresponds to a cue velocity of 2.75 m/s and '200' achieves a cue velocity 0.3 m/s. However, the velocity interval of 0.3-2.75 m/s is not divided into equal velocity intervals by the 200 strings explained above, but they do approximately divide the velocity range. However, all intermediate velocities are calculable using the values of the pulse widths used (the pulse widths are in the range of tens to hundreds of nanoseconds). This amount of resolution of the cue velocity can position the cue ball, theoretically, to a 15mm spatial accuracy on the table, but the repeatability characteristics of the robot, as described later, will also have an effect on the positioning accuracy.

For a given string sent from the PC, the driving signal of the servo should resemble a step input as far as the cue velocity is concerned, as the pulses are sent at a constant rate from the time t=0, which should theoretically ensure that the cue is moved at a constant speed from t=0. However, when the cue was tracked from the overhead camera and its speed was measured, the cue speed profiles were more like the plots shown in Figure 7.4. At lower cue speeds, the resulting cue speed almost follows the step-type input of pulses. However, for medium- to high-speed pulses, after an initial acceleration (which is due to the high friction present between the slides and linear guides) there is usually an overshoot in the velocity before it settles at the intended striking velocity (Figure 7.4). The settling time was also found to change with the intended velocities of the cue, the higher the desired velocity the higher the overshoot and settling time (or the cue travel distance in Figure 7.4). Hence, the ball was always

placed on the table at a distance of around 110mm, from the cue tip, in order to obtain consistent and stabilised strikes (110mm is greater than the settling distance for the maximum possible cue velocity as depicted in Figure 7.4).

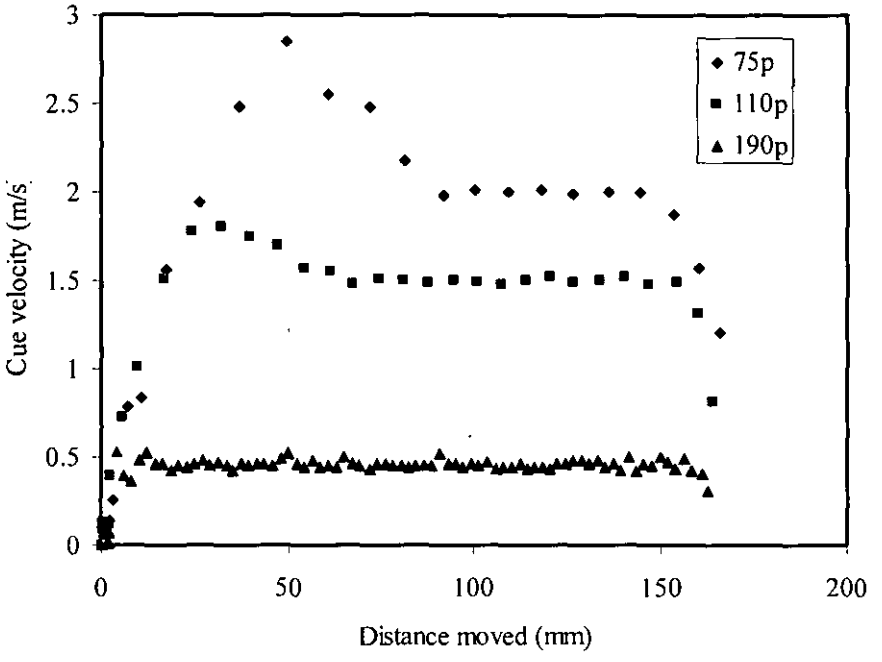


Fig. 7.4. The velocity variation of the cue against its travel for different control input codes for the servo driver

7.1.3.2 Repeatability of the Robot

To make consistent measurements with the robot and to compare and contrast the results that are obtained under different conditions, the robot's performance should be steady, i.e. it should be repeatable. The undesired dynamics of the mechanical setup, such as vibrations and inconsistencies in the electromechanical components, such as the servo used, can introduce inconsistencies into the system dynamics. Tests for repeatability also ensure that the electronics of the system, as in the case of sending pulses, also work steadily.

For these tests, the cue ball was kept at a particular position on the table and after each shot it was replaced back to that same spot on the table. In order to do this, two steel blocks were placed creating a wedge-shaped gap within which the cue ball was placed

repeatedly not touching the blocks (camera measurements have shown that the cue ball thus placed was within a tolerance of 0.2 mm). The cue was shot at a constant speed repeatedly for 10 times and each time the cue ball motion was tracked. Results were obtained for 3 sets of cue speeds: high, medium and low speed shots, with velocity codes of '90p', '150p' and '190p' respectively. No object ball was introduced on the table. The reason is that, unlike the cue ball, it was very difficult to place the object ball on the same spot on the table accurately, as any guide blocks cannot be introduced into the playing area into which the cue ball is shot. However, to introduce similar conditions to those present in ball-ball collisions, it was made sure that the cue ball collided with the cushion. The high-speed shot for code '90p', makes the ball collide 3 times with the cushions (the shots were taken almost across the table) and the ball travels for 3.5 m on the table. The ball under intermediate speed collides twice with cushions and has a total length of travel of around 2.4 m, and the low speeds shots make the ball have a single cushion collision with a 0.95 m distance of overall ball movement.

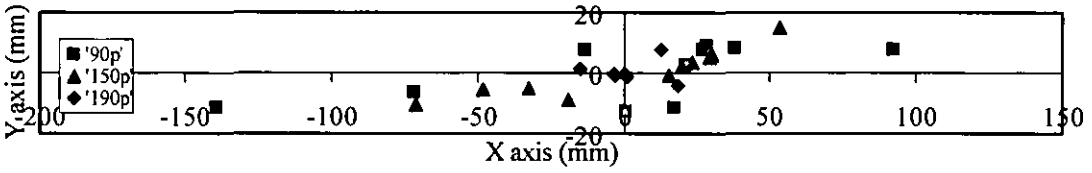


Fig. 7.5. Positioning results for 3 different speed shots around their respective mean values

The XY (i.e. the table coordinates) positioning results for the balls, when compared with the respective overall average of X and Y values for each individual shot, are plotted in Figure 7.5. For the single wall impact, low-speed shot, the positioning was done within ± 15 mm. Whereas for the high-speed shot, with 3 successive wall impacts, the ball was positioned within ±150 mm, which is approximately 3 times the ball diameter on each side of the mean. The reason for this deviation is that higher cue speeds lead to larger vibration amplitudes in the cue launcher. This vibration in turn affects the consistent hitting of the cue on the ball, and alters the hitting point. Although very small, this change will vary the amount of spin imparted to the ball, and this error propagates and becomes very large as the ball travels over longer distances and makes more collisions with the cushion. In a game of snooker, players mostly make use of the type of shot that is shown in Figure 7.1, the type of shots that consist of a cue ball-object ball collision and a cue ball-cushion collision. At other times, only a single collision between the cue ball and an object ball is employed. Players do use very complicated shots only in the absence of such simpler shots. Hence, the positioning repeatability of ±50 mm for the intermediate speed shot, which encounters two collisions and travels over a considerable length of 2.4 m, can be assumed for the robot as well. In summary, the repeatability of the robot is assumed to be equal to twice the ball diameter.

7.1.3.3 Cueing Dynamics

The cueing model proposed by de la Torre Juarez **[1994]** incorporates all of the effects that are present during cueing (see Section 2.4.1.1). The principal drawback with this model is in the determination of the impulse forces, especially in estimating the friction impulses between the cue and the cue ball. Even if the friction coefficient between the cue-tip and the ball is assumed to be known, leading to the calculation of the friction percussions, the directions of these percussions are difficult to determine due to the high deformability of the cue tip. This difficulty in the application of a theoretical model results in an experimentation-dependant approach.

For different cue offsets and cue speeds (V_{C0}, x'_0 and y'_0, as shown in Figure 7.3), the ball's motion was tracked by the overhead camera. In order to evaluate the initial ball spins, a single black pattern, such as the one used in Section 5.3, was kept upwards

facing the overhead camera. The theory presented in Section 5.4 was planned to be used to evaluate topspin and sidespin. This approach provides a mean to the determination of cueing dynamics as it happens in the real world.

The robot was initially set to stun conditions by adjusting the stepper drives in order to have x'_0=0 and y'_0=0. Five shots of different speeds at approximately 0.5 m/s intervals (in the range of 0.5 m/s to 2.8 m/s) were executed for the same ball position, by replacing the ball back to the initial position after each shot. Also, the cue was chalked well after each shot. For each shot the ball was placed such that the black pattern was within the view of the overhead camera and the ball motion was recorded at 180 fps (Figure 7.6). This very high-speed tracking introduces 'error ball speeds' of up to 0.05 m/s (sub-pixel level errors are magnified due to the finer resolution of time). Then y'_0 was varied from -12 mm to 12 mm in increments of 2 mm and for each y'_0, x'_0 was varied from zero to 12 mm also in 2 mm increments. For each combination of x'_0 and y'_0, 5 shots were played. Only right spin shots were played. As the cueing dynamics have a symmetry about the x'_0=0 line, the results obtained for right spins of the ball can be easily translated to left spins as well.

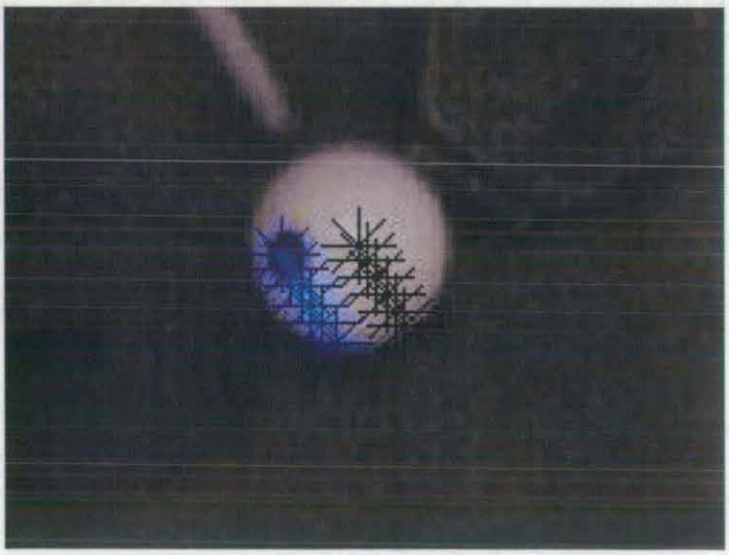


Fig. 7.6. Testing for the cueing dynamics, the pattern and the ball are tracked

The initial ball speed was measured as described in Chapter 5. Additionally, for the first few tracked points, an LMS error line was fit to obtain the direction of the ball movement as well. This direction was compared to the direction value obtained for

stun shots to obtain the deviation in the ball movement due to cue squirt. To calculate the initial ball spins, when calculations were performed according to the theory presented in Section 5.4, the values obtained for $\omega^T{}_0$ and $\omega^S{}_0$ were found to be totally unreasonable. For example, for shots with the slightest right-spin and zero topspin, the algorithm was found to give very large values for both top and right spins. This is possibly due to the drifting effect in the pattern-tracking that was found in Section 5.3, which was identified as a problem with the uneven lighting conditions present over the table area. This problem of estimating the initial ball spins is tackled in a different way in the coming sections.

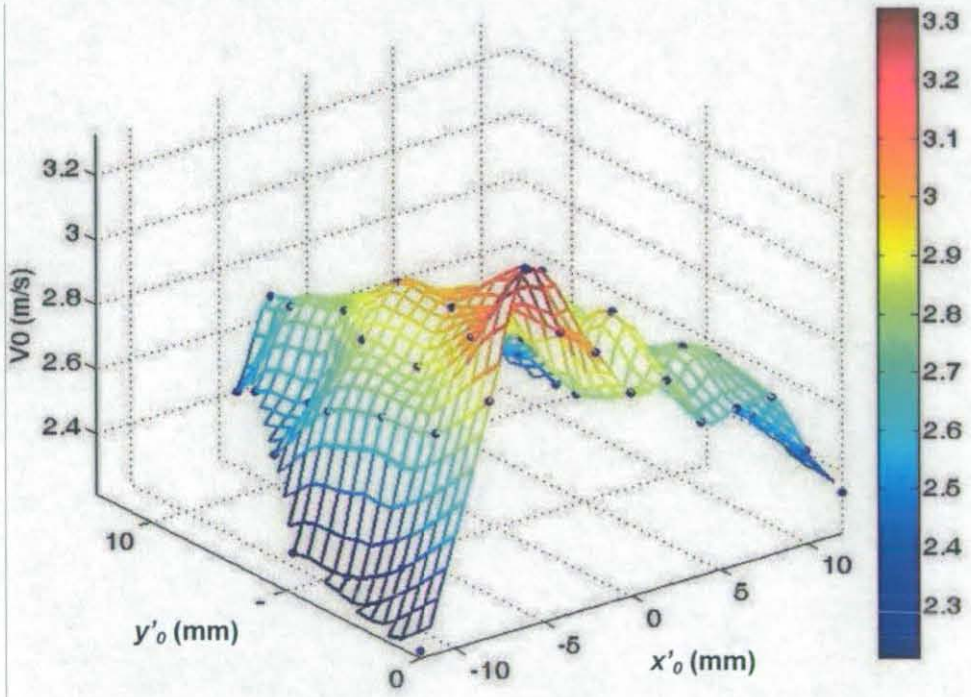


Fig. 7.7. Variation of the cue ball speed V_0 against x'_0 and y'_0 (measured in 2mm increments), for a cue speed corresponding to velocity control string '70p' (cue speed ~ 2.0 m/s)

Figure 7.7 shows the variation of V_0 with x'_0 and y'_0 for a cue velocity corresponding to the '70p' string sent to the servo drive. For such a constant cue speed, the cue ball's speed variation roughly resembles a dome having its apex around the point corresponding to a stun shot (i.e. $x'_0 = y'_0 = 0$). Hence, the velocity transfer from the cue to the ball degrades as the cue is hit further away from the 'stun' point (this should be compared with Figure 5.16). Cue squirt (α as shown in Figure 7.3) measurements for the same cue speed are shown in Figure 7.8. A maximum squirt value of around 2.5°

has been obtained. According to Figure 7.8, the squirt values almost remain the same when x'_0 is kept constant and y'_0 is changed. Hence, a general conclusion is that the squirt increases with x'_0, and does not depend greatly on the value of y'_0, which is easy to understand intuitively, and also after the experimental work of Cross **[2008]**, which suggests that squirt values for a chalked cue tip can go up to 10°. However, Cross' measurements were performed on a cue ball hung by a string. When the cue ball is on the table the resultant force of the normal and friction force components (denoted by T in Figure 2.11) at the cue-cue ball interface will be countered by the friction force at the cue ball-table interface, reducing the amount of squirt. Also, the amount of squirt depends on the properties of the cue shaft. Cross **[2008]** also concludes that thinner cue shafts can lead to reduced cue squirt because of their high flexibility rather than due to their reduced mass.

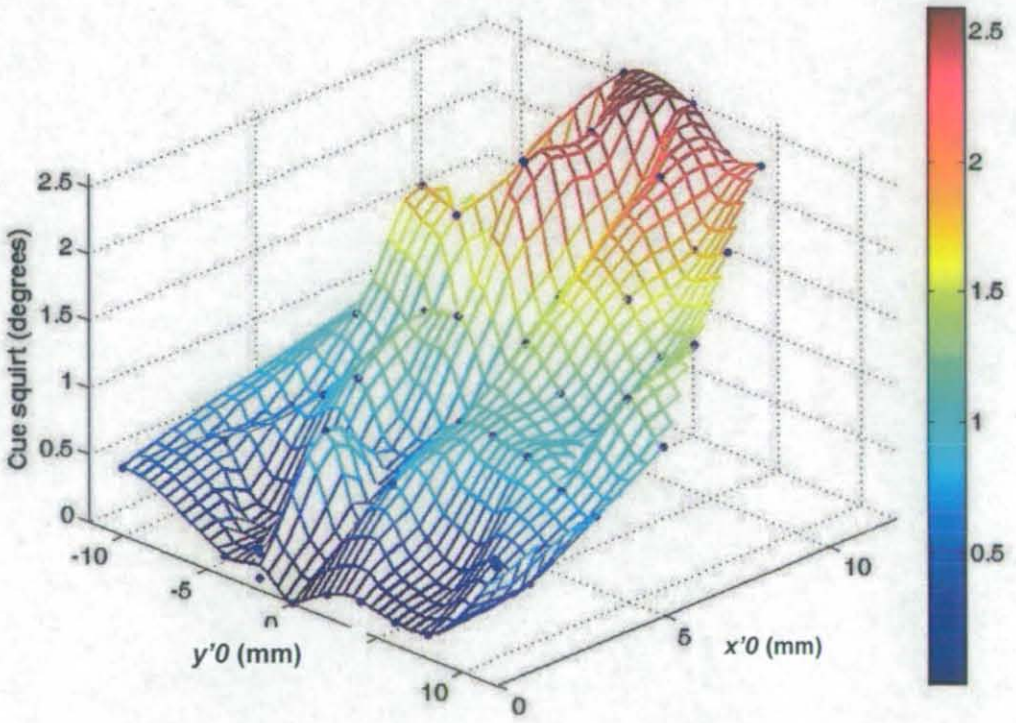


Fig. 7.8. Cue squirt values, α, against x'_0 and y'_0 (in 2mm increments), for a cue speed corresponding to velocity control string '70p'

These experimental results for cue ball speed and cue squirt have to be generalised so that for any given values of V_{C0}, x'_0 and y'_0 (also satisfying the constraints of equations (7.6) and (7.7)), the values of V_0 and α can be estimated. This generalisation of the dynamics can be performed in several ways. For example in a table tennis

robot, to generate an appropriate paddle movement according to the incoming conditions of the ball, an input-output map has been used **[Matsushima *et al.* 2005, Miyazaki *et al.* 2006]**. For this table tennis robot, empirically-obtained results are used to create the input-output maps of the dynamics, which essentially map out the outgoing ball conditions against respective incoming ball conditions for different paddle speeds and inclinations, and store them permanently in the memory of the controller of the robot. When a query is put forward, a Locally Weighted Learning (LWL) algorithm performs local interpolations, on the stored data, around the query point **[Miyazaki *et al.* 2006]**. However, the LWL method is memory expensive, as the complete set of empirical data has to be always retained in the system memory for retrieval. Artificial Neural Networks (ANNs) are also used in similar situations. For example Ming *et al.* **[2006]** use an ANN to derive the forward dynamics model of a golf swing robot. An advantage with ANNs is that once their training phase is over, the experimental data can be discarded, hence they are memory efficient.

Here, a neural network solution is proposed. ANNs consist of several artificial neurons (see Figure 7.9) that are arranged in different layers. A neuron j in the k^{th} layer of the network is connected to all neurons in the $(k\text{-}1)^{th}$ layer as well as in the $(k\text{+}1)^{th}$ layer, but does not have any connectivity to any other neurons in its own layer.

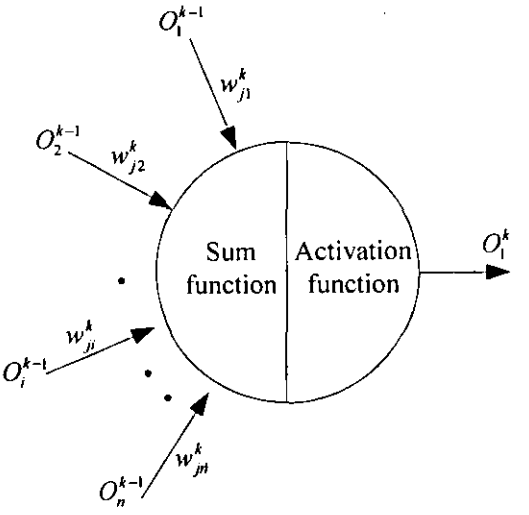


Fig. 7.9. An artificial neuron

In a neuron as shown in Figure 7.9, the summation function is calculated by, $net_j^k = \sum_j w_{ji}^k O_i^{k-1}$, where w_{ji}^k are known as connection weights. The activation function can be expressed as $O_j^k = f\left(net_j^k\right)$. A wide range of functions can be used for f, the activation function. Even within a network, different activation functions can be used for each of its layers.

Two backpropagation feedforward neural networks are trained to predict the cue ball speed, V_0, and cue squirt, α, separately for a given set of a cue velocity of V_{C0}, and cue offsets of x'_0 and y'_0. Feedforward networks are one of the most widely used models of ANNs, where the output of each node (also known as a neuron) propagates from the input side (left) to the output side (right) unanimously **[Jang *et al.* 1997]**, as shown in Figure 7.10. An error minimisation process usually trains the connection weights, w_{ji}^k, where for each neuron the Least Mean Square (LMS) error between the desired output T_j and the actual output O_j is estimated as $E = \frac{1}{2}\sum_j \left(T_j - O_j\right)^2$. In order to minimise the LMS error rapidly, an iterative error reduction of the gradient descent method (this is known as backpropagation, hence the name of the network) with an added momentum term is carried out in the following fashion, $\Delta w_{ji}^k(m+1) = \eta \frac{\partial E}{\partial w_{ji}^k} + \rho w_{ji}^k(m)$ where η is the learning rate, ρ is the momentum coefficient, and m is the index of iteration.

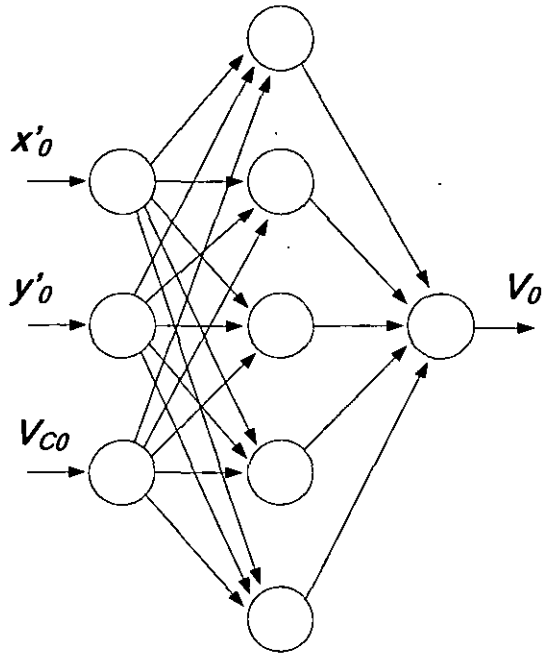


Fig. 7.10. A backpropagation feedforward neural network for predicting the value of V_0

Since the input data are of dimension 3, the input layer (the one on the extreme left in Figure 7.10) will have 3 nodes. The output of the network is essentially V_0, so the output layer has a single node. The network can have any number of hidden layers (the intermediate layers) with any number of nodes. No standard procedure exists that defines the exact number of layers and nodes to be used in the hidden layers for a given problem. Selecting the hidden layer configuration is a trial and error process, repeatedly evaluating the performance of the network. The neural networks are built in the Matlab® Neural Network Toolbox, which makes the processes of creating and training a network relatively simple. The prediction of the output value for any given input to the network is straightforward and performed with a simple command line option. The network uses the hyperbolic tangent function as the activation function (i.e. f) for its hidden layer and uses a linear activation function in its output layer **[Demuth and Beale 2001]**. When a set of empirical data is given to the network, the Toolbox sets aside some of the data for validation purposes and trains the network with the remaining data. A 3-5-1 neuron network was found to give good results in predicting both V_0 and α. The performance of the network trained for V_0 is shown in Figure 7.11, which also shows the curve for the validation process carried out in parallel with the training scheme. A trained network is stored in a MAT file format of

Matlab® and can be called by a single-line command from within any M-file. The two networks now represent the forward dynamic models for V_0 and α.

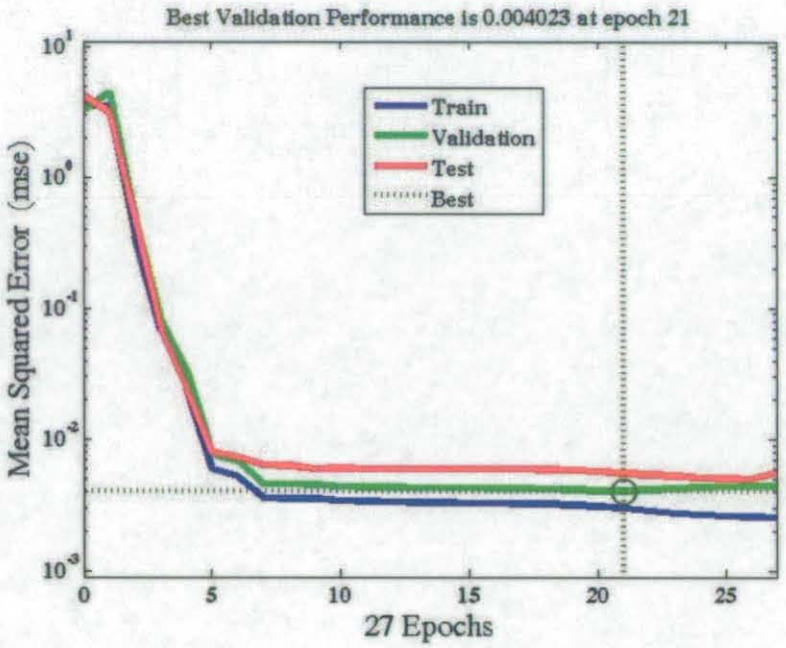


Fig. 7.11 Performance of the network for V_0 during its training phase

Estimating ω^T_0 and ω^S_0

Since the spin estimation using the pattern tracking did not provide reliable results, an alternative way had to be found to estimate ω^T_0 and ω^S_0. Researchers have often used the assumption of the cue tip gripping the cue ball during their impact **[Cross 2008, Shepard 2001, Alciatore 2004]**. This is largely owing to the fact the cue tip is well chalked before each shot, hence it has good frictional properties, and it is also flexible, hence it can easily deform as necessitated by the cue-cue ball interface dynamics. Being flexible and rough on its surface, the cue tip is assumed to grip the ball surface as soon as the cue and the cue ball come into contact. To provide the evidence for the instant gripping, Cross **[2008]** has performed a number of high-speed video-based observations on a cue striking a glass surface, and the movement of the cue tip at the interface. In fact, these high-speed camera-based measurements do show that the cue tip grips onto the glass surface immediately after both come into contact.

The assumption of instantaneous gripping as soon as the cue-tip touches the cue ball makes it reasonable to assume that the force transfer from the cue to the ball is in the direction of cue movement. Referring to Figure 7.3, the force component on the ball due to the cue striking will be only a resolved component of it (0.113 times the force, as $\sin\psi=\sin 6.5°=0.113$), which will give rise to a normal force at C. The friction force acting on the ball is even less, as the normal force at C is further multiplied by μ_s, estimated to be 0.21 in Section 5.1.1.2. When estimated, the friction force will only be 2% of the cue force. Moreover, it should also be noted here that de la Torre Juarez **[1994]** also suggests that when the cue is held almost horizontal, the frictional impulses from the table are negligible. The assumption of negligible frictional force between the ball and the table (and hence friction percussion) provides a means to estimate ${\omega^T}_0$ and ${\omega^S}_0$.

In the absence of friction forces, the effective value of the linear impulse transferred to the ball is $P_C = MV_0$. The coordinates of the cue hitting point were found to be $[x_C', y_C', z_C']$ in Section 7.1.2. As the cue impulse of P_C is the only impulse that is assumed to act on the ball, the following expressions for the initial values of sidespin and topspin can be written, denoting the moment of inertia of the ball by I ($I = 2MR^2/5$),

$$\omega_0^S = \frac{x_C' P_C}{I} = \frac{5x_C' V_0}{2R^2} \tag{7.8}$$

$$\omega_0^T = \frac{y_C' P_C}{I} = \frac{5y_C' V_0}{2R^2} \tag{7.9}$$

Nominally, right spin and topspin are treated as positive quantities according to the convention used in Figure 7.3.

7.1.4 Model for the Forward Dynamics

Figure 7.12 summarises the different elements involved in the forward dynamics of the ball for the situation shown in Figure 7.1.

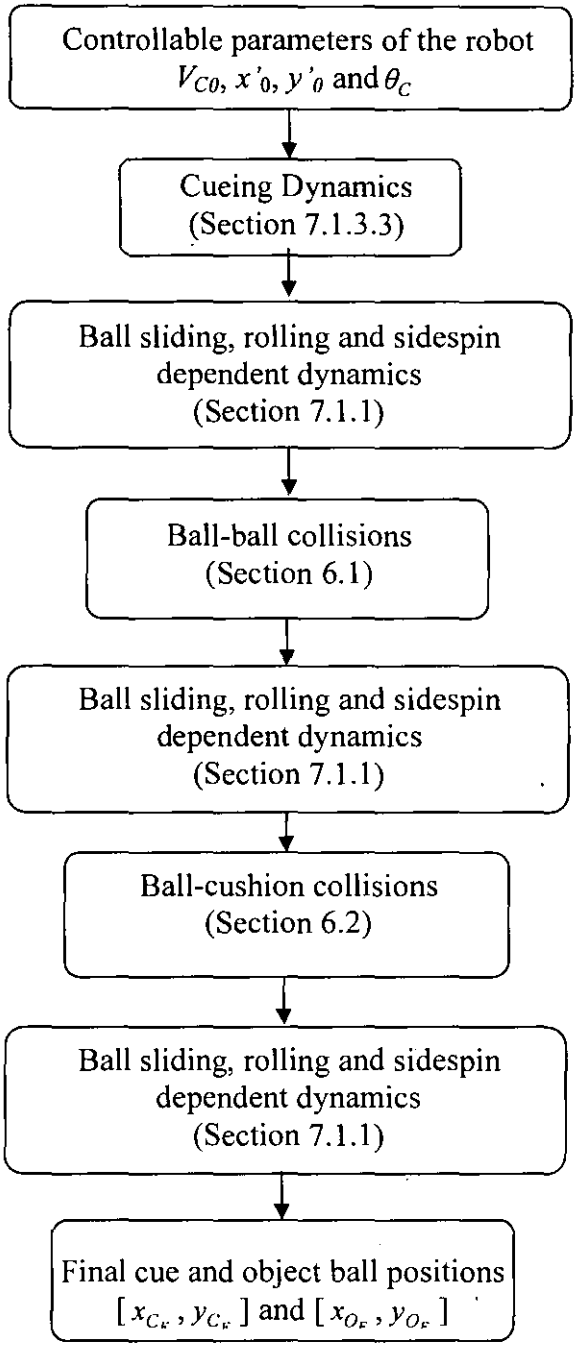


Fig. 7.12. Forward dynamics model

7.1.5 Definition of the Manipulation Problem

With reference to Figure 7.1, for a given cue ball location (C_I), targeted object ball (O_I) and pocket (P_I) combination, and to attain a certain desired final cue ball location (C_D), the task is to determine the initial required parameters of the ball motion, given by V_0, ω^T_0, ω^S_0, θ, and thereby establish the robot's manipulation parameters V_{C0}, x'_0, y'_0 and θ_C.

There can be other additional constraints such as other balls very close to the general area of trajectories that limit the possible ball trajectories, and hence the solution space, further. These constraints also have to be suitably tackled by the methodology that is used to obtain the solutions. However, the objective here is to obtain a solution for the fundamental manipulation problem defined above that is also depicted in Figure 7.1. Once a solution methodology is obtained, additional complexities such as additional balls in the vicinity can be treated as extended problems.

7.2 Manipulation Solutions

Now for all the interactions encountered in a regular shot, a form of descriptor for its dynamics is established. The solution for the positioning task (this also includes the potting of the object ball), set out in Section 7.1.4, is essentially an inverse one in nature. But some of the descriptors of dynamics are not explicitly expressed by equations, for example, the numerical solutions for the collisions and the empirical model derived for cueing. This complexity prevents one from obtaining the inverse solutions analytically. On top of this, there may arise situations where, due to the properties inherent to the dynamics of the system, an inverse solution does not exist. For example the object ball may not be able to be taken to C_D as required by the strategy-planning element, for any combination of manipulation parameters θ_C, x'_0, y'_0 and V_{C0}, so an option would be to take it to a position very close to C_D, for instance to C_F (Figure 7.1). Hence, the direct inverse solution based approach is ruled out. For positioning flat objects (axi-symmetric and polygonal ones) on a plane with the action of sliding friction, Huang *et al.* **[1995]** and Han and Park **[2001]** use inverse

numerical algorithms. However, the dynamics are very complicated in snooker and this approach cannot be used.

For a given positioning task, instead of finding a direct inverse solution, the manipulation space can also be searched by using the forward dynamics models. A possible solution can be found by trying to reduce the error in positioning, using a forward motion model of the object, whilst satisfying any possible constraints on the object motion. Various methodologies have been used in this regard. The major types of solutions used by various researchers are nonlinear optimisation **[Mason 1999, Li and Payandeh 2003b, Lynch and Black 2001]**, iterative learning control **[Zhu *et al.* 2006]** and machine learning **[Matsushima *et al.* 2005]**.

7.2.1 An Optimisation-based Solution

Here an optimisation-based approach is proposed to position the balls on the table. The optimisation function will have to be a composition of spatial errors between the actual positions where the balls will end up, and the desired ball locations. The conditions to ensure that the object ball is potted are also a part of the problem. This is generally known as nonlinearly constrained optimisation, and can be defined as **[Jang *et al.* 1997]**, referring to Figure 7.1,

For $q \in \Re^4$ and also subject to conditions in equations (7.2), (7.6) and (7.7),

$$\text{Minimise } F(q) = \left(x_{C_D} - x_{C_F}\right)^2 + \left(y_{C_D} - y_{C_F}\right)^2$$
$$\text{Subject to } [K(q)] \le [L] \text{ where, } q = [V_{C0}, x'_0, y'_0, \theta_C] \qquad (7.10)$$

The matrix condition $[K(q)] \le [L]$ consists of two elements. This constraint ensures that the object ball is potted by imposing conditions that the trajectory segment O_1O_F should go up to the pocket P_1 (or go past it) and that the minimum distance between the line segment and the centre of P1 must be less than 55 mm (for the ball to fall into the pocket). To use analytical-method-based optimisation, $F(q)$ has to be differentiable with respect to the robot's controllable variables q. Obtaining the

derivatives (i.e. the gradients) of $F(q)$ is not possible considering the difficulty in expressing the forward dynamics of the ball algebraically, in terms of q, as described earlier in this section. A numerical-based routine is still possible. Under similar conditions, a quasi-Newtonian method has been used by Li and Payandeh **[2003a]** for planar sliding objects and by Lynch and Black **[2001]** for a batting manipulator. In addition, many of the modern soft-computing methods such as Simulated Annealing, Genetic Algorithms, and Downhill Simplex Search are used for derivative-free optimisation **[Jang *et al.* 1997]**. Here Genetic Algorithms (GAs) are to be used.

7.2.1.1 Genetic Algorithms

GAs encode each point in the parameter space (or state-space, here denoted by q) into a binary bit string called a chromosome, and also associate each point with a fitness value, which is related to the optimisation evaluated at that point. For example, a state point q = [2.0m/s, 5rad, 4mm, 8 mm] of the robot can be represented by the following chromosome:

0010**0101**0100**1000**

GAs usually keep a set of points (chromosomes), instead of a single point, as a population, and this population is evolved continuously to a better overall fitness value. For each generation, a new population is constructed using genetic operations like selection, crossover and mutation. This evolution procedure is such that the chromosomes with better fitness values survive through to the next generation, and is based on Darwinian models of evolution **[Jang *et al.* 1997]**. The initial population is usually selected randomly.

Selection

After evaluation, a new population has to be created from the current population. The selection operation determines which chromosomes (also called parents) participate in the production of offspring for the next generation. In order to do this, parents are selected for mating with a selection probability proportional to their fitness values.

Crossover

By crossover, it is generally hoped that good features of the current generation will be retained in the successive generations, exploiting the current population's potential. Crossover is performed on selected pairs of parents with a probability equal to a given crossover rate. Figure 7.13 shows a single point crossover, where a crossover point on the parent chromosomes is selected at random and the chromosomes are interchanged at this point.

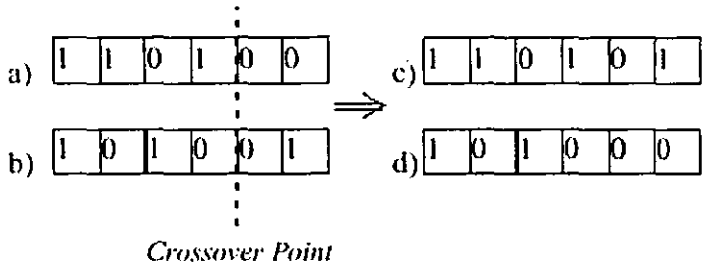


Fig. 7.13. Crossover operation with parents a and b and children c and d

Mutation

In case the current population does not have all the encoded information needed to solve the given optimisation problem, the crossover operator cannot lead to a satisfactory solution. On the other hand, a mutation operator that spontaneously generates new chromosomes can tackle this situation. The mutation operator is designed such that it flips a bit in the chromosome with a probability equal to a very low given mutation rate (Figure 7.14). The mutation rate is usually kept very low so that the good chromosomes obtained from the crossover operation are not lost.

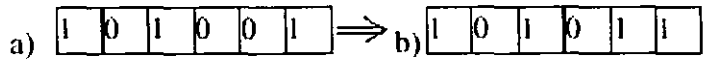


Fig. 7.14. Mutation operation

There are also various different modifications to the general process described above such as elitism, where certain best individuals from the current lot are passed on to the next generation without applying any of the above evolutionary processes.

The evolution process is repeated and the solution with best fitness for the optimisation is finally obtained.

7.2.2 Implementation of the GA-Based Optimisation

The GA-based optimisation is performed in the Matlab® Optimization Toolbox. The model describing the forward dynamics for the ball trajectory configuration shown in Figure 7.1 is programmed as an M-file function having q as its input. The numerical impact models of the ball-ball and the ball-cushion collisions (they themselves are M-file-based functions as described in Chapter 6) are called from within the forward dynamics M-file. The trained neural network for the cueing, the data of which are saved as a MAT file in Matlab®, is also used in the process. The straight-lines representing the cushions have also been mathematically established in the table coordinate system XY, using the overhead camera by placing the cue ball right next to the cushions. Similarly, the pocket centres have also been obtained. Also, various motions like straight-line sliding, rolling and curved motion profiles after collisions have been programmed. The instantaneous value of sidespin is also estimated before each collision. The constraints set out in equation (7.10) are also coded into the M-file.

The initial cue ball location is determined using the overhead camera as described in Chapter 5. A red ball is used as the object ball and its position on the table is established by processing the R component of the RGB colour image sequences obtained by the camera. These two parameters are embedded in the code. The desired final cue ball location is also specified. The M-file is then called from the Optimization Toolbox using its function handle and executed to deliver the best value for q. Also, various plots for the optimisation process can be readily obtained from the Toolbox.

7.3 Results

7.3.1 Tests on the Forward Dynamics Model

Initially, the validity of the forward dynamics model that was developed in this thesis was tested. If this was found to work well then the optimisation-based routine could be used to find the robot parameters for a given trajectory requirement set by the strategy subsystem. Initially the cue ball was placed before the robot and both the robot and the cue ball were adjusted to obtain stun conditions. The cue inclination was kept constantly at 6.5° with the horizontal. A shot was made (a stun shot) and the direction of movement of the ball, θ, was determined from the tracking performed by the camera. This procedure essentially establishes the orientation of the cue, θ_C, and if the robot has a swivel control unit (a motor-encoder combination) this process will not be needed.

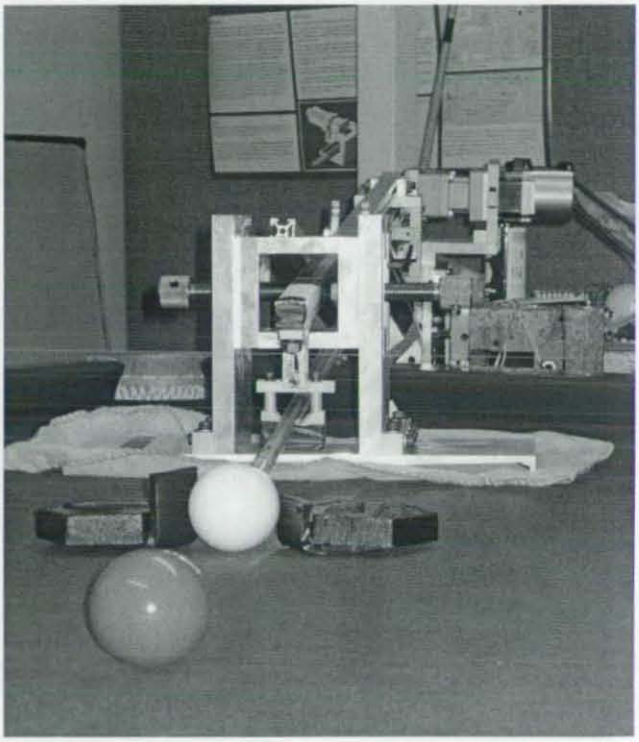


Fig. 7.15. The configuration of the system before a shot was taken

Now the steppers were activated to place the cue on the ball at a given location (x'_0 and y'_0). The frontal support was also manually adjusted according to the stepper

movement (Figure 7.15) and the front end of the cue holder was clamped. A velocity string was sent from the PC to activate the servo (with a velocity V_{C0}) and the overhead camera was triggered simultaneously to start tracking the motion of the balls (the camera was operated at 20fps because its ROI is a relatively large area). A typical tracked trajectory of the balls is given in Figure 7.16 (only the region of the table within the ROI settings is shown) where the cue ball also collides with the cushion once. The tracking algorithm provides the successive ball locations by processing every frame of the captured video. Hence, the final cue ball location and the equation for the final line of movement for the object ball can be determined in the table coordinate system XY.

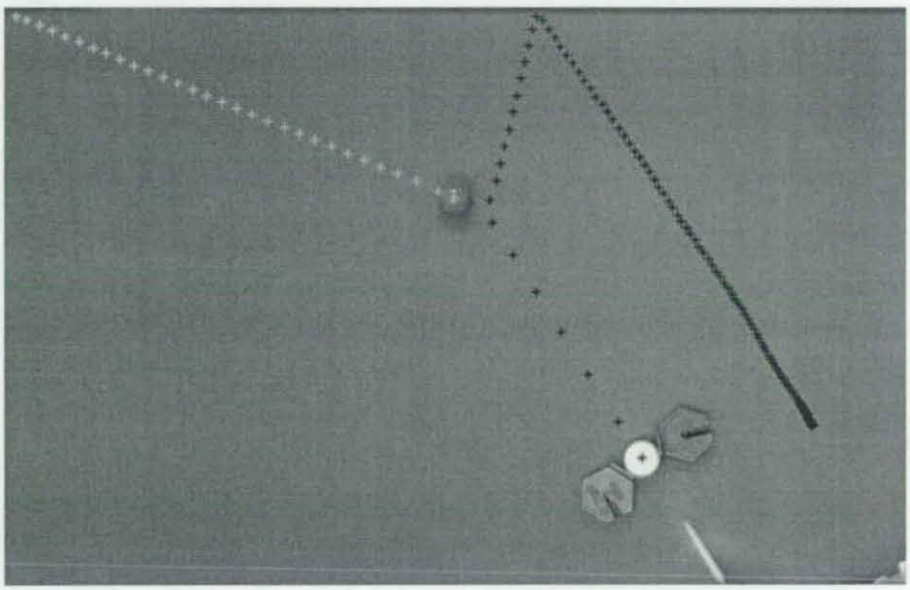


Fig. 7.16. A typical cue ball-object ball trajectory

Now the initial cue ball location, $[x_C, y_C]$, object ball location, $[x_O, y_O]$, the cue shift from the stun location (x'_0, y'_0) as set by the stepper drive, the cue orientation, θ_C, and its velocity V_{C0} are passed to the forward dynamics model to predict the final cue ball location and the final line of travel of the object ball. The line of the final object ball motion was also considered, as the camera can only view the top-half table area and the final object ball location could not always be obtained, as it may be outside the camera ROI. However, the final line of travel for the object ball determines the potting accuracy. Although it also has to go up to the pocket to fall in, this condition can basically be ensured by increasing the cue ball velocity.

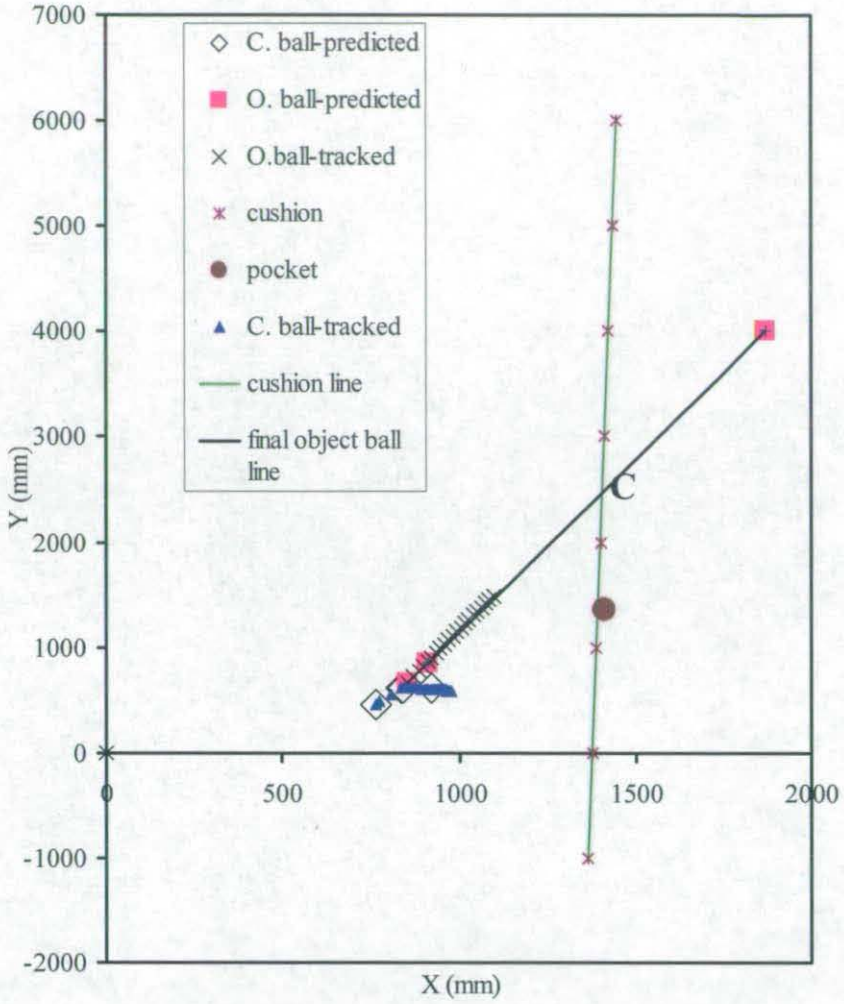


Fig. 7.17. Model predicted and experimentally obtained trajectories

Figure 7.17 shows the plot containing the spatial positions of the model-predicted and the camera-tracked values for the balls. The plot was obtained for a V_{C0} corresponding to the '140p' string, x'_0 = 6 mm and y'_0 = -6 mm. The predicted plots only have the initial ball positions, their final locations and the locations where the massé effect stops (the massé type trajectory of the ball is usually encountered after every collision). The cue inclination was 63.8° with the X-axis (measured by tracking a stun shot). It can be seen that the object ball is aimed at a random direction and it is not potted. The object ball can only be tracked up to a Y value of 1500 mm: at this distance the camera's field of view ends. The cue ball, in this case, was positioned to within 47mm of the predicted spot (this value is less than the ball diameter). For the object ball, to quantify the potting accuracy, the shortest distance between the actual path and the predicted path are evaluated at the location where the prediction curve meets the cushion (point C as shown in Figure 7.17). The cushion-predicted ball path

intersection was selected as a reasonable location for evaluation, as all the pockets are located along the cushions in snooker. For the shot shown above in Figure 7.17, the deviation between the two points at the cushion was found to be around 15 mm. As set out under the optimisation problem definition, this value can be up to 55 mm for snooker pockets for the ball to be pocketed, given that the final ball movement has adequate angles with the cushion for the ball not to be blocked by the cushion corners close to the pockets. Hence, for this shot the model prediction worked very satisfactorily.

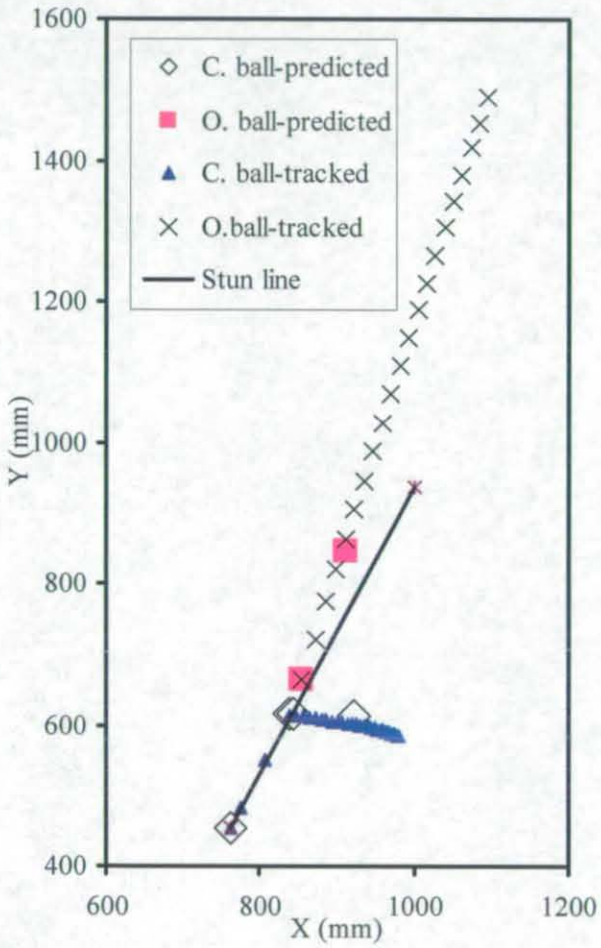


Fig. 7.18. Model-predicted and experimentally-obtained ball trajectories with the stun line (only a part of the prediction for the object ball is shown)

Figure 7.18 shows the same shot as in Figure 7.17, but more focussed onto the region of the observed ball trajectories, also showing the stun line, which is also the line of action of the cue (However, even this resolution is not enough to visualise the effects of cue squirt).

More than twenty shots were predicted and then experimentally evaluated (see Appendix II for some more results). Most of the time the object ball wall path was found to be within 50mm of its predicted value, which is quite acceptable in terms of the potting accuracy for the ball-pocket size configuration in snooker. Two attempts in the set had higher values, suggesting non-potting. This is probably due to the occasional inconsistencies in cueing, like a slight miscue, possibly due to vibration-induced effects in the robot. Also the repeatability characteristics of the robot, as evaluated in Section 7.1.3.2, could have had an influence over these inconsistencies. When it came to the positioning of the cue ball, positional error values in the range of 100-200mm were normally found.

With reference to Figure 7.1, if the next ball that is planned to be potted (O_2) is considerably far away (say, 750-1000 mm) from the intended (ideal) cue ball position (C_D), the positioning accuracy of 100-250mm will not greatly affect the next shot, as the angle of shot direction will not change drastically. However, if the next object ball is close to the planned cue ball location, this level of positioning accuracy can be a major concern, to the extent that the pre-planned object ball-pocket combination may not be possible anymore. A possible solution for this problem is to modify the strategy-planning element (the AI part) of the system so that this positioning error is taken into consideration when planning the shot sequence. Furthermore, the location of other balls in the vicinity of the intended cue ball location may also have an effect on the outcomes of potting the next ball in view of the trajectory errors.

Since the shots were played in a half table area, errors can be higher when the whole area of the table is used, as the shot lengths can be longer. The potting accuracy is still within acceptable limits, when the errors obtained for the half table area are doubled and compared with the potting requirements in snooker. However, the cue ball positioning errors did occasionally reach values considerably higher than 0.5 m. This could be attributed to the occasional high amplitude vibrations found when the robot was clamped insufficiently to its front support. The front support is shown in Figure 7.15.

7.3.2 Optimisation Algorithm Testing

As explained in Section 7.2, the coded M-file representing the forward ball dynamics is called from the Matlab® Optimization Toolbox. The M-file is also provided with the initial positions of the cue ball and the object ball; the left-side middle pocket is targeted in this experiment and its location is included in the M-file. A function tolerance value is used as the stopping criteria for the algorithm, where if the change in the optimised function value between two iterations is found to be less than the tolerance value set, the optimisation is terminated. The Toolbox is usually found to output the optimised values of θ_C, x'_0, y'_0 and V_{C0} in 3-4 minutes for a function (F in equation 7.10) tolerance of 25×10^{-6}. This function tolerance is equivalent to a circular area of 5 mm radius around the desired cue ball location C_D. A value of 5 mm was deemed sufficient, as even this level of accuracy is not possible from the robot (for a function tolerance of 10^{-6}, which is equivalent to positioning the cue ball within 1 mm, the optimisation time goes up to 40 min). Occasionally, multiple optimal values were also given, suggesting a multiple number of possibilities to obtain the specified positional results (for example the one with or without a cushion-cue ball impact).

The Optimization Toolbox also has the option of plotting various parameters during its search for the optimum solution. For example, Figure 7.19 is the plot of the variation of the average distance between members in a population plotted against successive generations during the search. The average distance is larger at the start of the search because the chromosomes are initiated throughout the search space and hence are spread out. Then, progressively, they all converge towards the optimum value; hence the final distance value is zero.

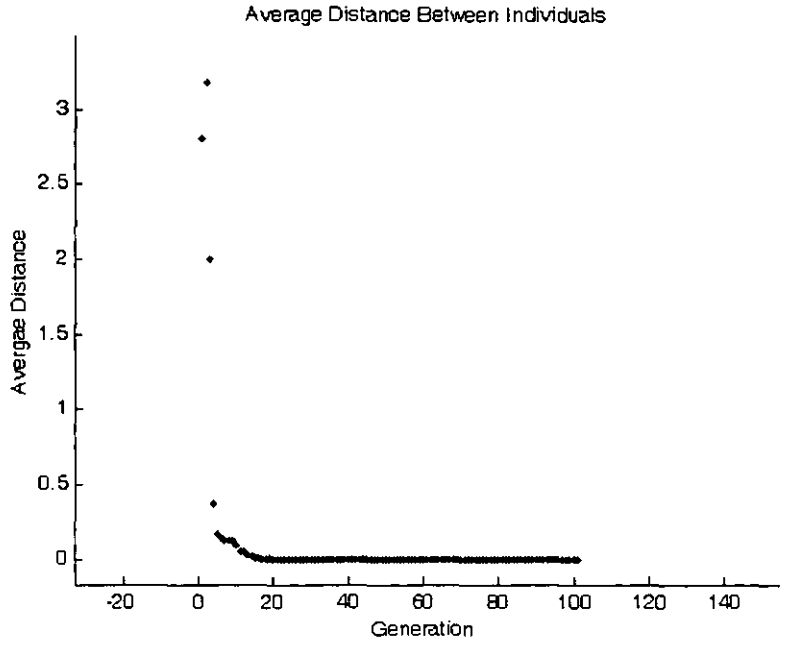


Fig. 7.19. Average distance between individuals in a population given for successive generations

For a θ_C obtained from the GA-based optimisation it is not always possible to set the cue launcher-mount accurately at that specific angle, since the table mount is very heavy and repositioning is found to be very cumbersome. In order to experiment with the results provided by the optimisation algorithm, the cue launcher was kept at a position for which the cue orientation θ_C was known, and this value was embedded in the code. Now the optimisation problem is of reduced dimensions (i.e. 3). The cue ball was placed on the table so that a stun shot could be executed by the robot. The object ball was kept at a place where it seemed that it had chances to be potted for the set cue orientation and cue ball position (the trajectory results for the object ball obtained in Section 6.1 were consulted in this regard). A target post-shot cue ball location was also provided to the algorithm.

For the following values of θ_C = 0.515 rad, $[x_C, y_C]$ = [698 mm, 562 mm], $[x_O, y_O]$ = [869 mm, 681 mm], and a desired cue ball location of $[x_{C_D}, y_{C_D}]$ =[1250 mm, 0 mm], the optimisation routine predicted the following parameters for the robot: V_{C0} corresponding to string '73p', x'_0 = -11 mm and y'_0 = 0 (x'_0 and y'_0 were approximated to the nearest millimetre).

The shot that was executed for the above results obtained from the GA optimisation is shown in Figure 7.20 (the pocket is not seen in the figure as it disappeared altogether once the image was undistorted). The ball was potted and the cue ball ended up at 110 mm from its desired location, $[x_{C_D}, y_{C_D}]$. The cue ball-cushion collision did not take place.

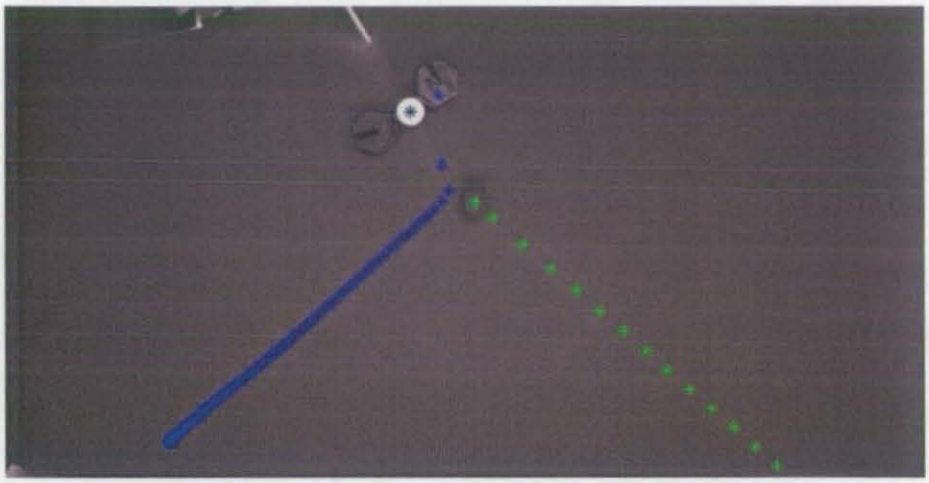


Fig. 7.20. Tracking results of the executed optimal shot

For the above results, the computer program representing the forward dynamics model was also executed and the intermediate slipping-rolling motion transition locations were also estimated as described earlier in this section. They are plotted together with the experimental values in Figure 7.21. These results make it possible to conclude that the above optimisation routine works well for this problem.

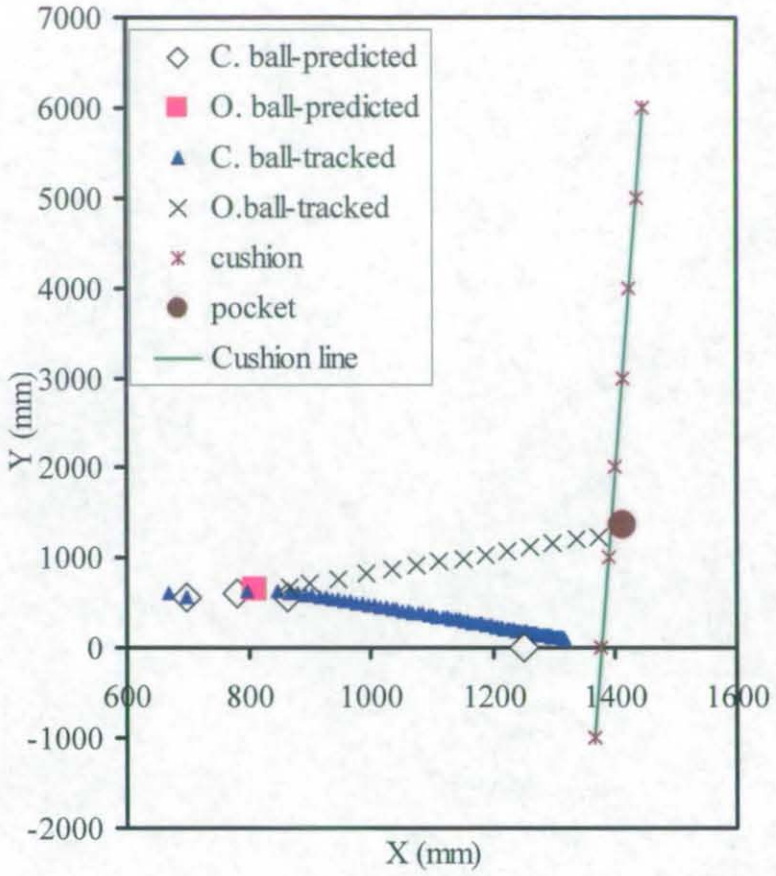


Fig. 7.21. Model predictions with experimental values for the optimal shot

Summary

In this chapter, the manipulation problem is defined with its appropriate background information and the relevant information needed from the earlier chapters is highlighted. A cueing model is established by combining experimental data with neural nets. An optimisation solution with a Genetic Algorithm search is suggested, programmed and finally tested for the positioning manipulation problem. The potting accuracy (this was estimated by calculating the prediction error at the cushion, as most of the shots were directed at random directions) was found to be very satisfactory over the half table area where the experiments were performed. Generally, a cue ball positioning accuracy in the range of 100-250 mm was also found.

Chapter 8

OVERALL DISCUSSION

This chapter discusses the relative merits/demerits of the developed system, approaches and methodologies.

8.1 System Design

The overall performance of the designed robotic system was found to be satisfactory as far as the scope of this project is concerned. The system has produced a wide-range of useful data that would be necessary for the design of an 'ultimate' robotic billiard/ snooker system, which was made possible by the very design of the system. However, when it comes to the overall design of a system that could move around the table (or over the table as in the case of gantry-based systems) a great deal of consideration must be given to the rack-and-pinion-based cue manipulation solution. A servomotor-driven rack and pinion solution has performed satisfactorily in closely controlling the cue velocity and in manipulating the cue like a human, unlike the pneumatic powered cues used in some other projects. However, the rotary to linear motion conversion is observed to produce very high moments in the system necessitating a very rigid reinforcement structure. This heavy structure, in turn, makes the system very heavy to move around the table. When the motor is actuated to drive the cue forward, the whole cue launcher sways on the vertical plane that contains the cue. This at times was found to lead to mis-hits, thus imparting a different spin to the ball, to that required from the system. This swaying motion is due to an active couple in the said vertical plane. The couple originates from the rotary to linear motion conversion at the gears. A frontal support for the cue launcher as seen in Fig. 7.15 minimised the sway, however, whenever the cue launcher was not adequately clamped, mis-hits were observed.

Recent videos of the Queen's University system also show that a shortened cue driven by a custom made linear actuator produces reasonably good ball speeds **[Deep Green**

2009]. As established in Section 5.1.4.1, the initial cue ball speed was only influenced by the cue velocity and not by the inertia coupled to the driving side of the cue. Also, considering the high flexibility of the cue, this raises doubts on whether the whole cue is taking part in the momentum transfer to the ball or whether only the frontal part of it actively drives the ball (Cross **[2008]** also makes an assumption to this effect when formulating a theoretical model for the cue squirt dynamics). If only a considerable front portion of the cue is found to actively participate in the momentum transfer (this can be proved only after extensive tests with shortened cues and then comparing their performances with the full-length ones), the cue may possibly be shortened, which again simplifies the cue-launcher configuration and makes it more compact. Hence, a shortened cue and a linear cue-driving configuration (without using a linear-rotary motion conversion) may prove to be more suited to a full-scale snooker robot capable of carrying the cue launcher easily around the table.

The stepper-based positioning system has lead to the manipulation of the cue ball with different types of spins, in order to take it to different regions on the table, and this is the first reported effort in that direction. Such a positioning system can also be replaced for its functionality by an eye-in-hand camera system and the employment of the visual servoing technique to precisely position the cue before striking the ball. Visual servoing may not be essential if a rigorous calibration procedure is performed on the robot, whereby the robot's position within the workspace (here the table area) can be accurately determined from the robot's joint encoder readings. However, robots having multiple serial links are often prone to positioning errors and require a visual servoing-based fine positioning arrangement.

The current system did not have a tilting facility to alter cue angles, and the cue inclination was kept constant at 6.5° to the horizontal. A fully-fledged robot will need this capability to deal with the balls that lie very close to the cushions as this situation requires a steep cue angle to the horizontal in order to make contact with the ball.

To improve the testing facilities with the current system (for a complete table testing procedure), a swivel actuator unit (a motor encoder combination) should be added to the robot. Without a swivel control unit the potting accuracy of the robot had to be

estimated at the cushion intersection of the object ball path, as it was cumbersome to manually and accurately set a swivel angle calculated by the forward dynamics model.

8.2 Computer Vision and Related Issues

The uneven lighting conditions present over the table area did not greatly affect the detection of the ball centres accurately, as the camera measurements were shown to be accurate and validated by physical measurements. This has enabled the extensive testing and reporting of the parameters found in the dynamics of snooker for the first time by a reliable methodology (the reported work in the American Journal of Physics is given in Appendix III). However, the efforts on the tracking of ball spin did not provide very reliable results. The existing fluorescent lighting over the table was found to produce excessive glare on the balls, leading to inconsistent detection of the pattern as it changed its position during the ball movement. The table resistance to the ball rotation about the vertical was measured by tracking the ball spin and this important value has been used in the forward model of the ball motion. However, three-dimensional spin-tracking is essential in establishing the cueing dynamics, as the existing models do not provide useful results. Thus an *in-situ* testing procedure is required.

The theories proposed in this thesis in relation to the impact dynamics of collisions also need an accurate spin-tracking element to enable their validation by the determination of ball spins prior to and after the impact.

8.3 Modelling of Collisions and other Dynamics

The model for the ball-cushion collision presented in this thesis is the first of its kind to perform a 3-dimensional analysis of the collision and predict the post-impact velocities and spin. Nevertheless, a further improvement to this model by a possible Finite Element Modelling of the area contact prevailing during the collision will make its predictions more accurate. When performing an analysis of the ball-cushion collision, a point contact was assumed between the cushion and the ball throughout

the time of impact. On the other hand, the high-speed motion-capture of the ball-cushion impulse performed by Alciatore **[2008]** and others reveals that the ball, in fact, deforms the cushion and makes contact over an area.

The assumption of decoupled motion (that between sidespin and the linear motion parameters like linear velocity and topspin) is based on the postulation of a point contact between the ball and the table. This assumption is obviously not the case in actuality, as the ball makes contact over an area on its surface; an evidence for the area contact is the presence of rolling friction, which cannot exist under the supposition of a point contact, as explained in Chapter 5. A detailed analysis on the influence of the area contact condition on ball motion will lead to a better understanding of the mechanics of rolling and sliding.

8.4 Manipulation Methods

This thesis explores the ways in which a snooker cue ball imparted with different spins can position itself and an object ball at different spatial positions on the table. For the first time, the connection between the standard, established, nonprehensile manipulation methodologies and the problem of spatial positioning present in snooker has been identified, and a relevant solution method has been proposed based on optimisation. Since the gradient-based optimisation methods cannot be used for this problem, a soft-computing based approach involving a Genetic Algorithm is employed. This optimisation procedure is found to take 3-4 minutes to converge to the optimal solution. Although this time is acceptable considering the slow nature of the game, some alternative optimisation procedures can also be considered. For example, Lynch and Black **[2001]** use a quasi-Newtonian nonlinear optimisation method for a single degree of freedom puck-juggler. According to Lynch and Black, this algorithm takes a few milliseconds to arrive at the optimal solution. Even though quasi-Newtonian nonlinear optimisation is a function-based method, and there are differences between the current problem and the one of Lynch and Black in terms of the estimation of the gradient of the optimisation function, a possibility for improvements in time is highlighted by the millisecond-level calculations reported by Lynch and Black.

Nonprehensile manipulation solutions are very useful in solving the problem of ball positioning in snooker. It is a programming or semi-programming-based approach where the knowledge about the dynamics of the object that is being manipulated is exploited to inform the robot on manipulation. There are alternatives to this approach. One such method is machine learning where an explicit knowledge about the system's dynamics is not given to the robot controller. Of late, machine-learning techniques are increasingly used in modern game-playing systems. An element of machine learning was used in establishing an Artificial Neural Network for the prediction of the cueing dynamics. An overall approach from a machine learning perspective should be able to correlate the final ball positions on the table with the robot control parameters, and should not involve the intermediate motion parameters like velocities and ball spins in its formulation. For example, Moore **[1991]** and Moore *et al.* **[1995]** implemented a memory-based learning scheme on the MIT robot. By only concentrating on the stun shots, and by changing the cue speed and the robot's swivel angle (a facility to impart different types of spin was not available in the robot), Moore *et al.* used 8 features, consisting of different length segments and various included angles of a ball s trajectory. All 8 features are stored in the robot's memory and when the robot is presented with a new situation, local function approximating techniques are used to find a solution. However, no efforts were made to position the cue ball. Moreover, the manipulation by imparting spin adds an additional complexity to the problem. With these additional complexities, the feature space could well be double the size of the feature space used by Moore *et al.* **[1995]**. To establish a strong correlation between the features and the solution space parameters (defined as q in Chapter 7), hundreds or possibly thousands of experiments may be required. This can only be performed on a fully automated system like the Queen's University robot **[Deep Green 2009]** and is not feasible with the experimental setup used in the present project, as it takes a longer time to move the system from place to place. Nevertheless, a breakdown of the overall problem into small sub-problems may prove to work with the present setup. For example, as performed for cueing, individual machine-learning-based models can be trained for the ball-ball collisions and the ball-cushion collisions with the aid of spin tracking.

8.5 Ball positioning performance

Experiments on ball positioning were performed within a table area of 5ft x 6ft. Within this area of the table, an object ball potting accuracy of more than 90% was obtained. In addition, the ball was positioned to an accuracy within the range of 100-250mm, in general. These are the first reported research efforts on the post-shot positioning of the cue ball. In its early stages of development, the Deep Green system was claimed to have 67% potting accuracy **[Long *et al.* 2004]]**; Deep Green plays on a pool table of size 4 ft x 8ft. However, the Deep Green research has not reported on the issues related to the cue ball positioning. In their latest publication, *Greenspan et al.* **[2008]** state that the robot has pocketed runs of four consecutive balls, but no quantitative figure is given for the ball potting accuracy. Here some facts concerning the pocket and the ball sizes in pool and snooker must also be considered. In snooker, all six pockets are 90 mm in size and the ball diameter is 52.5. If the mid-pocket entry is considered to be ideal for a object ball in snooker, the margin of maximum allowable error for a flawless entry (not touching the pockets) is around 19 mm, on either side of the ideal line of entry. However the way the cushion near the pocket entrance is shaped allows up to a 45 mm deviation for the corner pockets and a 55mm for the middle pockets, in snooker. Pool balls are 52.5 mm in diameter. In pool, the four corner pockets are 114-117 mm in size while the middle pockets measure 127-130 mm **[WPBA 2009]**. This leaves a robot with the margin of error of 28.5 mm for the corner pockets and 35 mm for the middle pockets, for a non contact-entry of the object ball; thus, the maximum possible values can also be expected to be larger than those in snooker. The preceding comparison underlines the fact that the ball potting is difficult in snooker. Another implication is, if the same robot is employed to play both the games, it will have a higher potting accuracy in pool when compared to that in snooker.

The performance of the current robot must be evaluated in light of other facts concerning the robot and the forward dynamics model for the ball motion. The robot's repeatability in ball positioning was found to be around ±50 mm and this, in turn, will unquestionably affect the positioning accuracy of the robot. In addition, a very basic model was utilized to estimate the initial values of sidespin and topspin of the ball immediately after cueing, using assumptions such as negligible friction from the table.

Hence, the forward model itself is not perfect. Suggestions have been made earlier in this chapter as to the improvements to the robot configuration that could possibly further reduce the vibration and the associated mishit problems. In addition, other suggestions in relation to spin tracking will lead to a more accurate model of cueing that is critically important for the success of a snooker robot.

Chapter 9

CONCLUSIONS AND FURTHER WORK

9.1 Conclusions

This thesis presents solution methodologies for the problem of manipulating snooker balls to achieve positional play. The following main research novelties have been achieved to present an overall solution to the problem.

9.1.1 Machine Vision

The capabilities of the overhead camera have been fully exploited, for the first time, to obtain the values of various physical parameters using accurate tracking at higher spatial and temporal resolutions. A broad range of tests on the dynamics of snooker has been performed and a journal paper based on these tests is accepted for publication in the American Journal of Physics. Some original efforts on ballspin tracking have been reported, and a ball with sidespin has been tracked using a single pattern. Furthermore, in relation to the tests on snooker dynamics, a thin film force sensor has been fixed to the snooker cue, so as not to affect its dynamics. Experiments with the force sensor and the overhead camera that tracks both the cue and the cue ball, have shown that the cue ball velocity is predominantly influenced by the cue velocity, regardless of the inertia driving the cue.

9.1.2 Analysis of Collisions in Snooker

An all-inclusive analysis of ball-ball collisions, incorporating all frictional effects, has been carried out without making any assumptions about the slip patterns at the impact point. Numerical analysis performed on the differential equations for the balls' motion, shows for the first time that the massé type spin will be imparted on the object ball due to the frictional forces at the collision point, and that the object ball will curve in its forward motion. A similar analysis has also been presented for the ball-cushion collision with the final solution being realised using numerical analysis.

9.1.3 Manipulation Methodologies

Using the results obtained from the high-speed camera and the numerical simulations of impacts, in conjunction with an empirical model of cueing, a forward model for the ball motion is put forward. A Genetic-Algorithm-based, gradient-free, optimisation procedure achieves a potting accuracy of over 90% and a cue ball positioning accuracy in the range of 100-250mm within a table area of 5 ft x 6 ft. The efforts on the cue ball positioning and the use of nonprehensile manipulation methods for this task are the first of their kind.

9.2 Recommendations for Further Work

The spin-tracking element has to be perfected, using very controlled lighting conditions, so that accurate spin transfer characteristics from the cueing operation can be determined. The present author believes that establishing the cueing dynamics accurately is crucial to the success of the project of creating a fully automatic snooker system, an aspect that has, hitherto, not been addressed in the literature in much detail. The theory on estimating sidespin and topspin of the ball over a limited area on the table by using a single overhead camera and a single circular pattern on the ball can be used. For the ball-cushion collision, the effects on the two making a surface contact must be investigated using Finite Element Models in order to achieve a better understanding of its dynamics. A spin-tracker will also validate the results of the collision models presented in this thesis.

The optimisation problem can possibly be reformulated giving lesser weight to the solutions where multiple impacts that may give rise to additional uncertainties will be present. For example, a forward solution with only a cue ball-object ball impact should be given an advantage over a solution where the cue ball makes an additional impact with the cushion before positioning itself to the desired location.

REFERENCES

Alciatore, D. G. (2004). *The Illustrated Principles of Pool and Billiards*. ed. New York: Sterling Publishing. ISBN: 1-4027-1428-9.

Alciatore, D. G. (2008). High-speed Pool and Billiards Video Clips. http://www.billiards.colostate.edu/high_speed_videos/, last accessed 15 July 2009.

Alciatore, D. G. (2009). Pool and Billiards Physics Principles by Coriolis and Others. submitted to the American Journal of Physics, manuscript number: MS22090. http://billiards.colostate.edu/physics/Alciatore_AJP_MS22090_revised_pool_physics_article.pdf, last accessed 20 January 2009

Alian, M. E., Shouraki, S. E., Shalmani, M. T. M., Karimian, P., and Sabzemeydani, P. (2004). Roboshark: A Gantry Pool Player. 35th International Symposium on Robotics(ISR). Paris, France. http://ce.sharif.ac.ir/~karimian/works/publications/fullpaper223ebnealian.pdf, last accessed 15 January 2009.

Andersson, R. L. (1989). Dynamic Sensing in a Ping-Pong Playing Robot. *IEEE Transactions on Robotics and Automation* 5(6), pp: 728-739. ISSN: 1042-296X.

Asada, M., Kitano, H., Noda, I., and Veloso, M. (1999). RoboCup: Today and Tomorrow-What we have learned. *Artificial Intelligence* 110(2), pp: 193-214. ISSN: 1362-3079.

Bayes, J. H., and Scott, W. T. (1962). Billiard-Ball Collision Experiment. *American Journal of Physics* 3(31), pp: 197-200. ISSN: 0002-9505.

Bishop, B. E., and Spong, M. W. (1999a). Adaptive Calibration and Control of 2d Monocular Visual Servo Systems. *Control Engineering Practice* 7(3), pp: 423-430. ISSN: 0967-0661.

Bishop, B. E., and Spong, M. W. (1999b). Vision-Based Control of an Air Hockey Playing Robot. *Control Systems Magazine, IEEE* 19(3), pp: 23-32. ISSN: 0272-1708.

Bouguet, J. Y. (2008). Camera Calibration toolbox for MATLAB®. http://www.vision.caltech.edu/bouguetj, last accessed 18 January 2008.

Campbell, M., Hoane, A. J., and Hsu, F.-H. (2002). Deep Blue. *Artificial Intelligence* 134(1-2), pp: 57-83. ISSN: 0004-3702.

Candea, C., Hu, H., Iocchi, L., Nardi, D., and Piaggio, M. (2001). Coordination in multi-agent RoboCup teams. *Robotics and Autonomous Systems* 36(2-3), pp: 67-86. ISSN: 0921-8890.

Cheng, B.-R., Li, J.-T., and Yang, J.-S. (2004). Design of the Neural-Fuzzy Compensator for a Billiard Robot. *2004 IEEE International Conference on Networking, Sensing & Control*, 21-23 March 2004 Taipei, Taiwan. pp: 909-913. ISBN: 0780381939.

Chua, S. C., Wong, E. K., and Koo, V. C. (2005). Intelligent Pool Decision System Using Zero-Order Sugeno Fuzzy System. *Journal of Intelligent and Robotic Systems* 44(2), pp: 161-186. ISSN: 0921-0296.

Chua, S. C., Wong, E. K., and Koo, V. C. (2007). Performance Evaluation of Fuzzy-Based Decision System for Pool. *Applied Soft Computing* 7(1), pp: 411-424. ISSN: 1568-4946.

Chua, S. C., Wong, E. K., Tan, W. C., and Koo, V. C. (2002). Decision Algorithm for Pool Using Fuzzy System. *Proceedings of the International Conference on Artificial Intelligence in Engineering & Technology (iCAiET 2002)*, 17-18 June 2002 Kota Kinabalu, Malaysia, pp: 370-375.

Chua, S. C., Wong, E. K., Koo, V. C. (2003). Pool Balls Identification and Calibration for a Pool Robot. *Proceedings of the International Conference on Robotics, Vision, Information and Signal Processing (ROVISP 2003)*, 22-24 January 2003 Penang, Malaysia. pp: 312-315.

Clauser, C. E., McConville, J. T., and Young, J. W. (1969). Weight, Volume, and Center of Mass of Segments of the Human Body. Technical Report. Ohio, Aerospace Medical Research Laboratory, Aerospace Medical Division, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio. http://www.dh.aist.go.jp/bodyDB/m/e-k-01.html, last accessed 7 March 2009.

Click-Snooker. http://www.clicksnooker.co.uk/, last accessed on 5 August 2008.

Cross, R. (2005). Bounce of a spinning ball near normal incidence, *American Journal of Physics*, 73(10), pp: 914-92. ISSN: 0002-9505.

Cross, R. (2008). Cue and ball deflection (or "squirt") in billiards. *American Journal of Physics* 76(3), pp: 205-212. ISSN: 0002-9505.

Davis, K. (2009). A Watchful Eye. http://www.matrox.com/imaging/en/press/feature/other/hawkeye/, last accessed 02 August 2009.

de la Torre Juarez, M. (1994). The Effect of Impulsive Forces on a System with Friction: The Example of the Billiard Game *European Journal of Physics* 15(4), pp: 184-190. ISSN: 1361-6404.

Demuth, H., and Beale, M., (2001). Matlab Neural Network Toolbox User's Guide. www.mathworks.com/access/helpdesk/help/pdf_doc/nnet/nnet.pdf, last accessed 1 June 2009.

Denman, H., Rea, N., and Kokaram, A. (2003). Content-Based Analysis for Video from Snooker Broadcasts. *Computer Vision and Image Understanding* 92(2-3), pp: 176-195. ISSN: 1077-3142.

Deep Green (2009). http://www.deepgreenrobot.org/, last accessed 25 June 2009.

Domenech, A. (2008). Non-smooth modelling of billiard- and superbilliard-ball collisions. *International Journal of Mechanical Sciences* 50(4), pp: 752-763. ISSN: 0020-7403.

Domenech, A., and Casasus, E. (1991). Frontal Impact of Rolling Spheres. *Physics Education* 26(3), pp: 186-189. ISSN: 0031-9120.

Domenech, A., Domenech, T., and Cebrian, J. (1987). Introduction to the Study of Rolling Friction. *American Journal of Physics* 55(3), pp: 231-235. ISSN: 0002-9505.

D'Orazio, T., Guaragnella, C., Leo, M., and Distante, A. (2004). A New Algorithm for Ball Recognition Using Circle Hough Transform and Neural Classifier. *Pattern Recognition* 37(3), pp: 393-408. ISSN: 0031-3203.

Dussault, J.-P., and Landry, J.-F. (2006). Optimization of a Billiard Player - Position Play. *LECTURE NOTES IN COMPUTER SCIENCE* 4250, pp: 263-272. ISSN: 0302-9743.

Dussault, J.-P., and Landry, J.-F. (2007). Optimization of a Billiard Player - Tactical Play. *LECTURE NOTES IN COMPUTER SCIENCE* 4630, pp: 256-270. ISSN: 0302-9743.

Fogel, D. B., and Chellapilla, K. (2002). Verifying Anaconda's Expert Rating by Competing against Chinook: Experiments in Co-Evolving a Neural Checkers Player. *Neurocomputing* 42(1), pp: 69-86. ISSN: 0925-2312.

Gennery, D. B. (2006). Generalized Camera Calibration Including Fish-Eye Lenses. *International Journal of Computer Vision* 68(3), pp: 239-266. ISSN: 0920-5691.

Gratton, L. M., and Defrancesco, S. (2006). A Simple Measurement of the Sliding Friction Coefficient. *Physics Education* 41(3), pp: 232-235. ISSN: 0031-9120

Greenspan, M. (2005). Uofa Wins Pool Tournament. *International Computer Games Association Journal* 28(3), pp: 191-193. ISSN: 1389-6911.

Greenspan, M. (2006). Method and Apparatus for Positional Error Correction in a Robotic Pool System Using a Cue-Aligned Local Camera. Canadian Patent, Number: CA 2520923.

Greenspan, M., Lam, J., Godard, M., Zaidi, I., and Jordan, S. (2008). Toward a Competitive Pool-Playing Robot. *Computer* 41(1), pp: 46-53. ISSN: 0018-9162.

Griffiths, I., Evans, C., and Griffiths, N. (2005). Tracking the Flight of a Spinning Football in Three Dimensions. *Measurement Science and Technology* 16(10), pp: 2056-2065. ISSN: 0957-0233.

Grogono, P. (2005). The Snooker Simulation. http://users.encs.concordia.ca/~grogono/Snooker/snooker.html, last accessed 25 June 2009.

Han, I., and Park, S.-U. (2001). Impulsive Motion Planning for Positioning and Orienting a Polygonal Part. *The International Journal of Robotics Research* 20(3), pp: 249-262. ISSN: 0278-3649.

Hashimoto, K., and Noritsugu, T. (1996). Modelling and Control of Robotic Yo-Yo with Visual Feedback. *Proceedings of 1996 IEEE International Conference on Robotics and Automation*, 22-28 April 1996, Minneapolis, MN, USA. pp: 2650-2655. ISBN: 0-7803-2988-0.

Heikkila, J., and Silven, O. (1997). A Four-Step Camera Calibration Procedure with Implicit Image Correction. *1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, San Juan, Puerto Rico IEEE Computer Society, pp: 1106-1113. ISBN: 0-8186-7822-4.

Hierrezuelo, J., and Carnero, C. (1995). Sliding and Rolling: The Physics of a Rolling Ball. *Physics Education* 30(3), pp: 177-182. ISSN: 0031-9120.

Ho, K. H. L., Martin, T., and Baldwin. J. (2007). Snooker Robot Player - 20 Years On. *IEEE Symposium on Computational Intelligence and Games (CIG 2007)*, 1-5 April 2007, Honolulu, Hawaii. pp: 1-8. ISBN: 1-4244-0709-5.

Hopkins, D. C., and Patterson, J. D. (1977). Bowling Frames: Paths of a Bowling Ball. *American Journal of Physics* 45(3), pp: 263-266. ISSN: 0002-9505.

Huang, W. H., Krotkov, E. P., and Mason, M. T. (1995). Impulsive Manipulation. *IEEE International Conference on Robotics and Automation*, 21-27 May 1995 Nagoya, Japan. IEEE. pp: 120-125. ISBN: 0-7803-1965-6.

Huang, W. H., and Mason, M. T. (2000). Mechanics, Planning and Control for Tapping. *The International Journal of Robotics Research* 19(10), pp: 883-894. ISSN: 0278-3649.

Jang, J.-S. R., Sun, C.-T., and Mizutani, E. (1997). *Neuro-Fuzzy and Soft Computing - a Computational Approach to Learning and Machine Intelligence*. First ed. New Jersey: Prentice Hall. ISBN: 0-13-261066-3.

Jebara, T., Eyster, C., Weaver, J., Starner, T., and Pentland, A. (1997). Stochasticks: Augmenting the Billiards Experience with Probabilistic Vision and Wearable Computer. *Proceedings of the IEEE International Symposium on Wearable Computers*, Oct 1997 Cambridge, MA, USA, pp: 138-145. ISBN: 0-8186-8192-6.

Jin, H.-L., and Zacksenhouse, M. (2004). Robotic Yoyo Playing with Visual Feedback. *Robotics, IEEE Transactions on* 20(4), pp: 736-744. ISSN: 1552-3098.

Jewett, B. (1994). Squirt Testing. *Billiards Digest.* Luby Publishing, Inc., Chicago, Illinois, August 1994, pp: 34-38.

Kane, T. R., and Levinson, D. A. (1987). An explicit solution of the general two-body collision problem. *Computational Mechanics* 2(1), pp: 75-87. ISSN: 1432-0924.

Komi, E. R., Roberts, J. R., and Rothberg, S. J. (2007). Evaluation of thin, flexible sensors for time-resolved grip force measurement. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science* 221(12), pp: 1687-1699. ISSN: 0954-4062.

Lam, J., Long, F., Roth, G., and Greenspan, M. (2006). Determining Shot Accuracy of a Robotic Pool System. *Proceedings of the 3rd Canadian Conference on*

Computer and Robot Vision (CRV'06), 7-9 June 2006, Quebec City, Canada. IEEE Computer Society. pp: 23-30. ISBN: 0-7695-2542-3.

Lam, J., and Greenspan, M. (2008). Eye-In-Hand Visual Servoing for Accurate Shooting in Pool Robotics. *Proceedings of the 2008 Canadian Conference on Computer and Robot Vision (CRV'08)*, 28-30 May 2008, Ontario, Canada. IEEE Computer Society. pp: 11-17. ISBN: 0-7695-2542-3.

Landry, J.-F., and Dussault, J.-P. (2007). AI Optimization of a Billiard Player. *Journal of Intelligent and Robotic Systems* 50(4), pp: 399-417. ISSN: 0921-0296.

Larsen, L. B., Jensen, M. D., and Vodzi, W. K. (2002). Multi Modal User Interaction in an Automatic Pool Trainer. *Proceedings of the Fourth IEEE International Conference on Multimodal Interfaces (ICMI'02)*, 14-16 Oct, 2002 Pittsburgh, USA. pp: 361-366. ISBN: 0-7695-1834-6.

Leckie, W., and Greenspan, M. (2005). Pool Physics Simulation by Event Prediction 1: Motion Transitions. *International Computer Games Association Journal* . 28(4), pp: 214-222. ISSN: 1389-6911.

Leckie, W., and Greenspan, M. (2007). Monte Carlo Methods in Pool Strategy Game Trees. *LECTURE NOTES IN COMPUTER SCIENCE* 4630, pp: 244-255. ISSN: 0302-9743.

Li, Q., and Payandeh, S. (2003a). Multi-Agent Cooperative Manipulation with Uncertainty: A Neural Net-Based Game Theoretic Approach. *Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on*, 14-19 Sept 2003 Taipei, Taiwan. IEEE. pp: 3607-3612. ISBN: 0-7803-7736-2.

Li, Q., and Payandeh, S. (2003b). Planning Velocities of Free Sliding Objects as a Free Boundary Value Problem. *The International Journal of Robotics Research* 23(1), pp: 69-87. ISSN: 0278-3649

Lin, Z. M., Yang, J. S., and Yang, C. Y. (2004). Grey Decision-Making for a Billiard Robot. *Systems, Man and Cybernetics, 2004 IEEE International Conference on* 10-13 October 2005. pp: 5350-5355. ISBN: 0-7803-8566-7.

Long, F., Herland, J., Tessier, M.-C., Naulls, D., Roth, A., Roth, G., and Greenspan, M. (2004). Robotic Pool: An Experiment in Automatic Potting. *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 28 Sept-2Oct 2004 Sendal, Japan. pp: 2520-2525. ISBN: 0-7803-8463-6.

Lynch, K. M., and Black, C. K. (2001). Recurrence, Controllability, and Stabilization of Juggling. *IEEE Transactions on Robotics and Automation* 17(2), pp: 113-124. ISSN: 1042-296X.

Lynch, K. M., and Mason, M. T. (1999). Dynamic Nonprehensile Manipulation: Controllability, Planning, and Experiments. *The International Journal of Robotics Research* 18(1), pp: 64-92. ISSN: 0278-3649.

Marlow, W. C. (1994). *The Physics of Pocket Billiards*. ed. Florida: MAST. ISBN: 0-9645370-0-1.

Mason, M. T. (1999). Progress in Nonprehensile Manipulation. *The International Journal of Robotics Research* 18(11), pp: 1129-1141. ISSN: 0278-3649.

Matsushima, M., Hashimoto, T., and Miyazaki, F. (2003). Learning to the Robot Table Tennis Task - Ball Control and Rally with a Human. *Systems, Man and Cybernetics, 2003. IEEE International Conference on*, 5-8 October 2003 Washington DC, USA. IEEE. pp: 2962-2969. ISBN: 0-7803-7952-7.

Matsushima, M., Hashimoto, T., Takeuchi, M., and Miyazaki, F. (2005). A Learning Approach to Robotic Table Tennis. *IEEE Transactions on Robotics* 21(4), pp: 767-771. ISSN: 1552-3098.

Medwell, B., Price, A., and Velleman, J. (2004). Robotic Pool Player. Project Report, Robotics Research Group, University of Adelaide. http://www.mecheng.adelaide.edu.au/robotics_novell/projects/2002/Pool/final.pdf, last accessed 12 July 2009.

Ming, A., Furukawa, S., Teshima, T., Shimojo, M., and Kajitani, M. (2006). A New Golf Swing Robot to Simulate Human Skill-Learning Control Based on Direct Dynamics Model Using Recurrent Ann. *Mechatronics* 16(7), pp: 443-449. ISSN: 0957-4158.

Ming, A., and Kajitani, M. (2003). A New Golf Swing Robot to Simulate Human Skill-Accuracy Improvement of Swing Motion by Learning Control. *Mechatronics* 13(8-9), pp: 809-823. ISSN: 0957-4158.

Miyazaki, F., Matsushima, M., and Takeuchi, M. (2006). Learning to Dynamically Manipulate: A Table Tennis Robot Controls a Ball and Rallies with a Human Being. *Advances in Robotic Control BOOK CHAPTER- SPRINGER (Editors Kawamura, S., Svinin, M.) pp: 317-341. Spinger Berlin Heidelberg. ISBN:* 978-3-540-37347-6.

Moore, A. W. (1991). Fast, Robust Adaptive Control by Learning Only Forward Models. *Advances in Neural Information Processing Systems 4 (NIPS Conference)*, December 2-5, 1991 Denver, Colorado, USA. Morgan Kaufmann. pp: 571-578. ISBN: 1-55860-222-4.

Moore, A. W., Hill, D. J., and Johnson, M. P. (1995). An Empirical Investigation of Brute Force to Choose Features, Smoothers and Function Approximators. *Computational Learning Theory and Natural Learning Systems* 3: Selecting Good Models, pp: 361--379. ISSN: 0-262-58133-7.

Nadler D (2005). *Mathematical Theory of Spin, Friction, and Collision in the Game of Billiards*. ed. California, David Nadler. ISBN-10: 0977167100. (Translation of Coriolis, G. –G. (1835), *Théorie Mathémathique des Effets du Jeu de Billard.*, ed. Paris, Carilian-Goeury)

Neilson, P. J., Jones, R., Kerr, D., Sumpter, C. M. (2004). An image recognition system for the measurement of soccer ball spin characteristics. *Measurement Science and Technology* 15, pp: 2239-2247. ISSN: 0957-0233.

Onoda, G. Y. (1989). Comment On "Analysis of Billiard Ball Collisions in Two Dimensions" By R.E.Wallace and M.C.Schroeder[Am.J.Phys.56,56,815-819(1988)]. *American Journal of Physics* 57(5), pp: 476-478. ISSN: 0002-9505.

Park, J. I., Partridge, C. B., and Spong, M. W. (2001). Neural Network-Based State Prediction for Strategy Planning of an Air Hockey Robot. *Journal of Robotic Systems* 18(4), pp: 187-196. ISSN: 0741-2223.

Partridge, C. B., and Spong, M. W. (2000). Control of Planar Rigid Body Sliding with Impacts and Friction. *The International Journal of Robotics Research* 19(4), pp: 336-348. ISSN: 0278-3649.

Pingali, G., Opalach, A., and Jean, Y. (2000). "Ball tracking and virtual replays for innovative tennis broadcasts," *Proceedings of the 15th International Conference on Pattern Recognition* (vol. 4), Barcelona, Spain, 3-7 September 2000, pp. 152-156. ISBN: 0-7695-0750-6.

Rezzoug, N., and Gorce, P. (1999). Dynamic Control of Pushing Operations. *Robotica* 17(6), pp: 613-620. ISSN: 0263-5747.

Ronsse, R., Lefevre, P. and Sepulchre, R. (2006). Sensorless Stabilization of Bounce Juggling. *IEEE Transactions on Robotics* 22(1), pp: 147-159. ISSN: 1552-3098.

Russell, S., and Norvig, P. (2005). *Artificial Intelligence - A Modern Approach* Second ed. Delhi: Pearson Education, Inc. ISBN: 81-297-0041-7.

Salazar, A., and Sanchez-Lavega, A. (1990). Motion of a Ball on a Rough Horizontal Surface after Being Struck by a Tapering Rod. *European Journal of Physics* 11, pp: 228-232. ISSN: 0143-0807.

Senoo, T., Namiki, A., and Ishikawa, M. (2004). High-Speed Batting Using a Multi-Jointed Manipulator. *Proceedings of 2004 IEEE International Conference on Robotics and Automation.* , 26/04/2004-01/05/2004 New Orleans, USA. IEEE. pp: 1191-1196. ISBN: 0-7803-8232-3.

Shepard, R. (1997). Amateur Physics for the Amateur Pool Player. http://www.tcbilliards.com/articles/physics.shtml, last accessed 25 June 2009.

Shepard, R. (2001). Everything you always wanted to know about squirt, but were afraid to ask. http://www.sfbilliards.com/shepard_squirt.pdf, last accessed 20 January 2009.

Shu, S. W. (1994). *Automating Skills Using a Robot Snooker Player*. PhD thesis, University of Bristol.

Smith, M. (2006a). *Pickpocket: An Artificial Intelligence for Computer Billiards*. (Master's Thesis). University of Alberta, Alberta.

Smith, M. (2006b). Running the Table: An AI for Computer Billiards. *Proceedings of The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference*, 16-20 July 2006 Boston, Massachusetts, USA. AAAI Press. pp: 994-999. ISBN: 978-1-57735-281-5.

Smith, M. (2007). PickPocket: A computer billiards shark. *Artificial Intelligence* 171(16-17), pp: 1069-1091. ISSN: 0004-3702

Spong, M. W. (2001). Impact Controllability of an Air Hockey Puck. *Systems & Control Letters* 42(5), pp: 333-345. ISSN: 0167-6911.

Stronge, W. J. (2000). *Impact Mechanics*. ed. Cambridge: Cambridge University Press. ISBN: 0-521-63286-2.

Tekscan (2009). Tekscan, Inc. http://www.tekscan.com/, last accessed 02 April 2009.

Thornton, C., and Ning, Z. (1998). A theoretical model for the stick/bounce behaviour of adhesive, elastic-plastic spheres. *Powder Technology* 99(2), pp: 154-162. ISSN: 1568-5527.

Voyenli, K. and Eriksen, E. (1985). On the Motion of an Ice Hockey Puck. *American Journal of Physics* 53(12), pp: 1149-1153. ISSN: 0002-9505.

Walker, J. (1983). The Physics of the Follow, the Draw and the Masse (in Billiards and Pool). *Scientific American*. New York, USA, Scientific American, Inc. (USA). **249**, pp: 124-129. ISSN: 0036-8733.

Wallace, R. E., and Schroeder, M. C. (1988). Analysis of Billiard Ball Collisions in Two Dimensions. *American Journal of Physics* 56(9), pp: 815-819. ISSN: 0002-9505.

Wang, Y., and mason, M. T. (1992). Two-Dimensional Rigid-Body Collisions With Friction. *Journal of Applied Mechanics* 59(3), pp: 635-642. ISSN: 0021-8936.

Williams, K. (2002). *Know the Game Snooker*. 3rd ed. London: A&C Black(Publishers) Ltd. ISBN: 0-7136-6001-5.

Witters, J., and Duymelinck, D. (1986). Rolling and Sliding Resistive Forces on Balls Moving on Flat Surface. *American Journal of Physics* 54(1), pp: 80-83. ISSN: 0002-9505.

WPBA (2009). World Pool-Billiard Association. http://www.wpa-pool.com/, last accessed 15 July 2009.

WPBSA(2008). World Professional Billiards & Snooker Association. http://www.worldsnooker.com/, last accessed 26 May 2009.

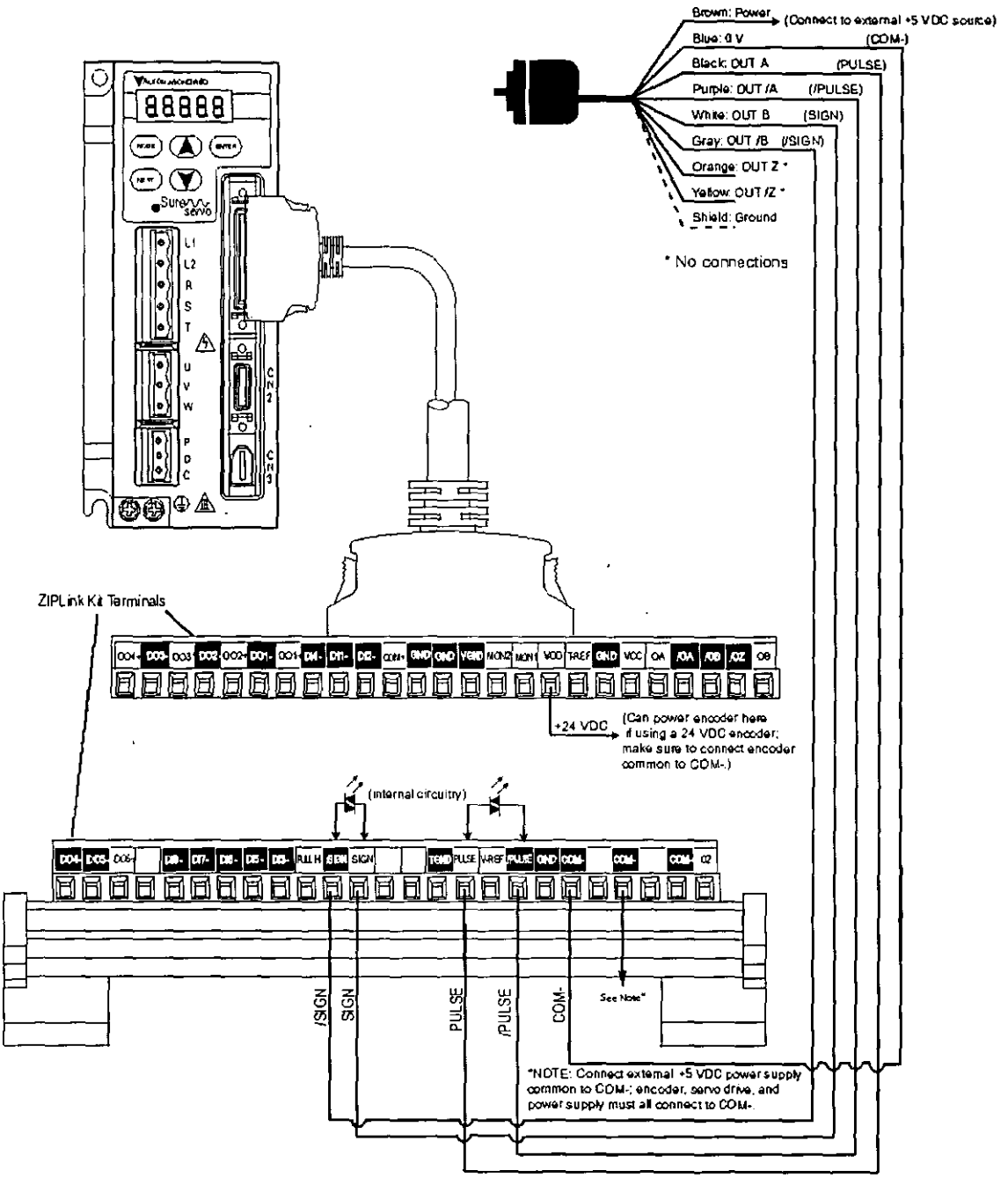
Zhang, X., and Vu-Quoc, L. (2002). Modelling the dependence of the coefficient of restitution on the impact velocity in elasto-plastic collisions. *International Journal of impact Engineering* 27(3), pp: 317-341. ISSN: 0734-743X.

Zhang, Z. (2000). A Flexible New Technique for Camera Calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(11), pp: 1330-1334. ISSN: 0162-8828.

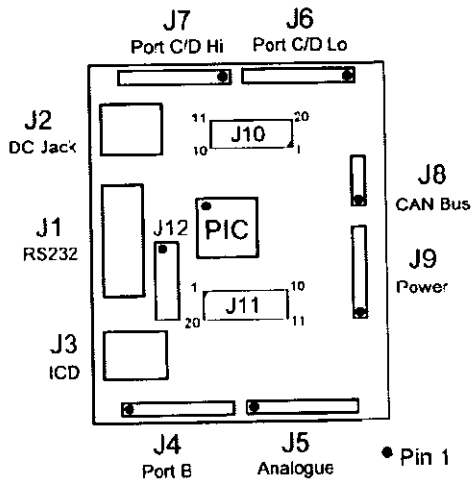
Zhu, C., Aiyama, Y., Arai, T., and Kawamura, A. (2006). Positioning in Releasing Manipulation by Iterative Learning Control. *Journal of Intelligent and Robotic Systems* 46(4), pp: 383-404. ISSN: 0921-0296.

APPENDIX I: HARDWARE DETAILS

CONTROL OF *SureServo*™ SERVOMOTOR BY EXTERNAL PULSES



GENERAL PROGRAMMING BOARD (IENSYS® BOARD INCORPOARATED WITH A PIC 18F458 MICROCONTROLLER)



J1 – RS232 Socket

1	NC
2	TXD
3	RXD
4	DSR
5	GND
6	DTR
7	CTS
8	RTS
9	NC

J2 –DC Power In

Outer GND
Inner 9-25V dc smoothed

J3 – uCHIP ICD

1	MCLR/VPP
2	+5V
3	GND
4	RB7/PGD
5	RB6/PGC
6	RB5/PGM

J4 – Port B

1	GND	
2	RB0/INT0	
3	RB1/INT1	
4	RB2/CANTX	
5	RB3/CANRX	
6	RB4	
7	RB5/PGM	J12, 3
8	RB6/PGC	J12, 4
9	RB7/PGD	J12, 5
10	Vcc (+5v)	

J5 – Analogue

1	GND
2	RA0/AN0
3	RA1/AN1
4	RA2/AN2
5	RA3/AN3
6	RA5/AN4
7	RE0/AN5
8	RE1/AN6
9	RE2/AN7
10	Vcc (+5v

J6 – Port C/D Lo

1	GND
2	RC0/T1OSO
3	RC1/T1OSI
4	RC2/CCP1
5	RC3/SCK
6	RD0/PSP0
7	RD1/PSP1
8	RD2/PSP2
9	RD3/PSP3
10	Vcc (+5v)

J7 – Port C/D Hi

1	GND
2	RC4/SDI
3	RC5/SDO
4	RC6/TX
5	RC7/RX
6	RD4/P1A
7	RD5/P1B
8	RD6/P1C
9	RD7/P1D
10	Vcc (+5v)

J8 – CAN Bus

1	CAN In Hi
2	CAN In Lo
3	CAN Out Hi
4	Can Out Lo

J9 – Power & Reset

1	Vdd
2	Vdd
3	GND
4	GND
5	Vcc
6	Vcc
7	RA4/T0CKI (1WB)
8	MCLR

J10 - Expansion

1	Vcc (+5v)
2	RC0/T1OSO
3	RC2/CCP1
4	RD0/PSP0
5	RD2/PSP2
6	RC4/SDI
7	RC6//TX
8	RD4/P1A
9	RD6/P1C
10	Vdd (+12v)
11	GND
12	RD7/P1D
13	RD5/P1B
14	RC7/RX
15	RC5/SDO
16	RD3/PSP3
17	RD1/PSP1
18	RC3/SCK
19	RC1/T1OSI
20	RA6

J11 - Expansion

1	GND
2	RB0/INT0
3	CANH
4	RB5/PGM
5	RB7/PGD
6	RA0/AN0
7	RA2/AN2
8	RA4/T0CKI (1WB)
9	RE0/AN5
10	RE2/AN7
11	GND
12	RE1/AN6
13	RA5/AN4
14	RA3/AN3
15	RA1/AN1
16	MCLR
17	RB6/PGC
18	RB4
19	CANL
20	RB1/INT1

J12 – JUMPERS *[default]*

1	CTS Disconnect – removal allows use of RC4 *[absent – no CTS handshake]*
2	RX Disconnect – removal allows use of RC7 *[fitted – RS232 active]*
3	RB5 – allows use of port when ICD not in use *[absent – ICD in use]*
4	RB6 – allows use of port when ICD not in use *[absent – ICD in use]*
5	RB7 – allows use of port when ICD not in use *[absent – ICD in use]*

PIXELINK CAMERA PL-B776

Specifications

Colour / Mono	Colour
Resolution	2048 x 1536
Frame Rate at Full Resolution	12
Sensor Type	CMOS
Shutter Type	Rolling
Lens Format	C1/2"
Pixel Pitch	3.2 μm
Sensor Diagonal	8.19 mm
Bit Depth	8 or 10
Power Consumption (Watts)	3.2W
Variable ROI	Yes
Right-angle Capable	Yes
Interface (FireWire)	6 pins x 2

Camera Features via FireWire

Trigger Options	Hardware - Optically Isolated 5-12V @ 4-11mA, Software and Free Running
General Purpose Outputs	2 Optically Isolated - Maximum 40V Differential. Maximum 15mA

Image Quality Measures

Responsivity (Peak)	1.8DN/(nJ/cm^2)
Dynamic Range	60 dB
FPN	< 1%
PRNU	< 1%
Read Noise	< 1 DN

Camera Calibration Results

The values of intrinsic parameters generated for the current camera setup are given below. Refer to http://www.vision.caltech.edu/bouguetj/calib_doc/ or Heikkila and Silven **[1997]**, for an explanation of the concepts.

Intrinsic Parameters:

Focal length in pixels, fc = [$fc(1)$ $fc(2)$] = [1798.6, 1797.8]

Principal point coordinates, cc= [$cc(1)$ $cc(2)$] = [1070.4, 738.9]

Skew coefficient defining the angle between the x and y pixel axes, $alpha\_c = 0$

Image distortion coefficients (radial and tangential coefficients),

kc = [-0.3363, 0.1408, 0.0006, 0.0012, 0].

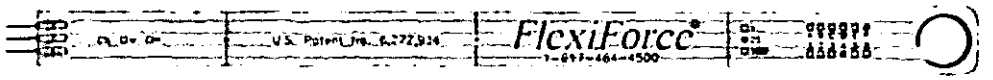
Extrinsic parameters:

$$Rc = \begin{bmatrix} 0.009334 & 0.999954 & 0.002338 \\ 0.999825 & -0.009371 & 0.016206 \\ 0.016227 & 0.002186 & -0.999866 \end{bmatrix}, \quad Tc = \begin{bmatrix} -559.748458 \\ -613.678549 \\ 1710.132676 \end{bmatrix}$$

Where the values of elements of Tc are in mm and the camera coordinate frame and the world frame OXYZ are related by the equation,

$$XX_c = Rc * XX + Tc$$

FlexiForce® FORCE SENSOR



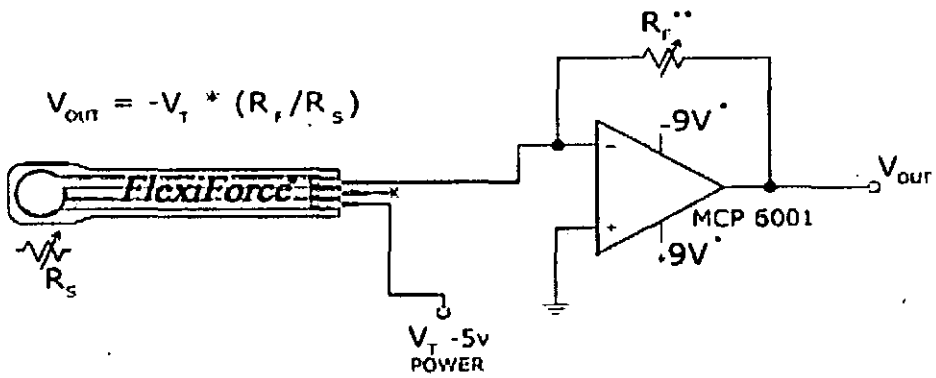
FlexiForce A201 Sensor

Specifications

Thickness	0.208 mm
Length	203 mm
Width	14 mm
Connector	3-pin male connector
Linearity error	<± 3%
Repeatability	<± 2.5%
Hysteresis	< 4.5%
Drift	< 5%
Response time	< 5 µs
Operating Temperature	-9 °C to 60 °C
Force range	0-440 N*
Temperature sensitivity	Output variance up to 0.36% per °C

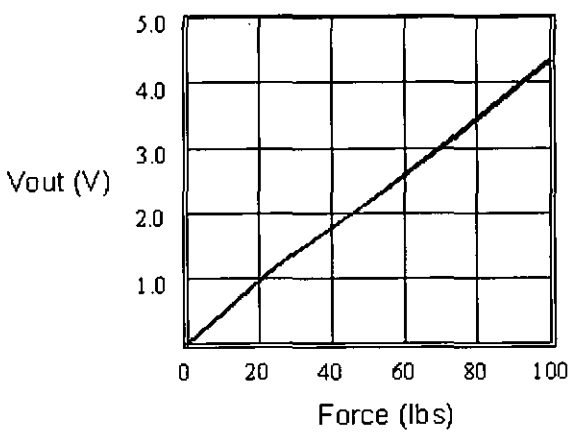
* To measure above 440 N apply a lower drive and reduce the resistance of the feedback resistor (1 kΩ minimum; see driving circuit)

Driving Circuit

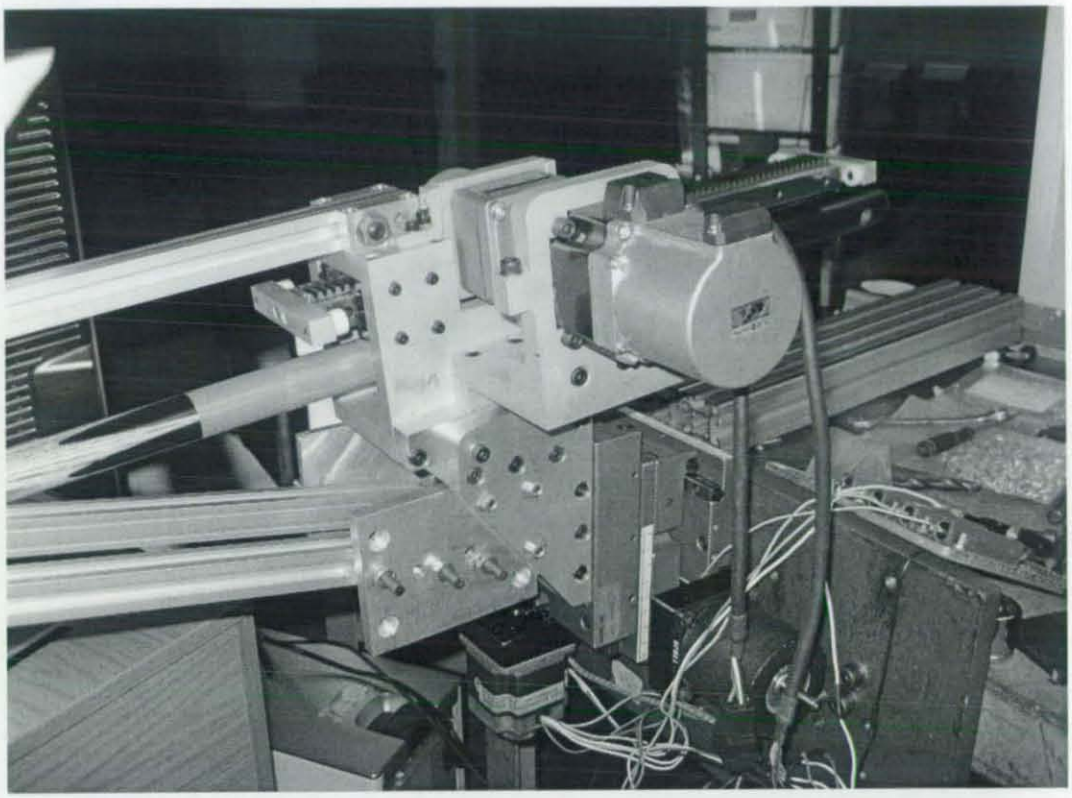


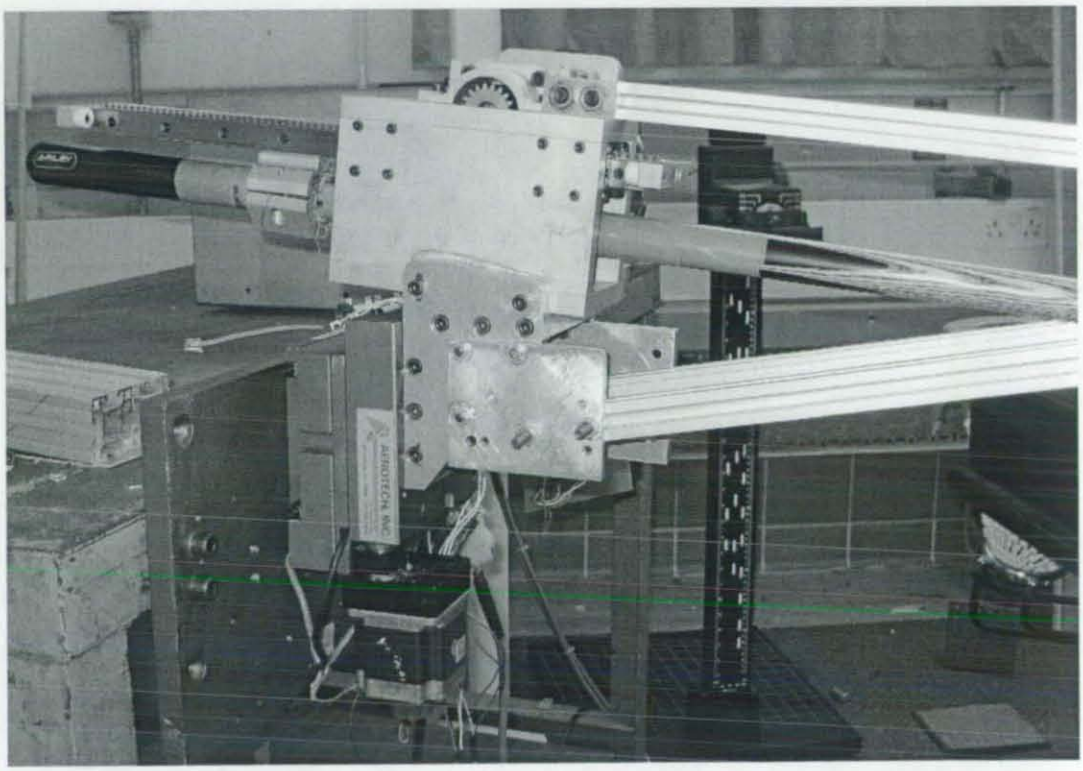
- * Supply Voltages should be constant
- ** Reference Resistance R_F is 1kΩ to 100kΩ
- Sensor Resistance R_S at no load is >5MΩ
- Max recommended current is 2.5mA

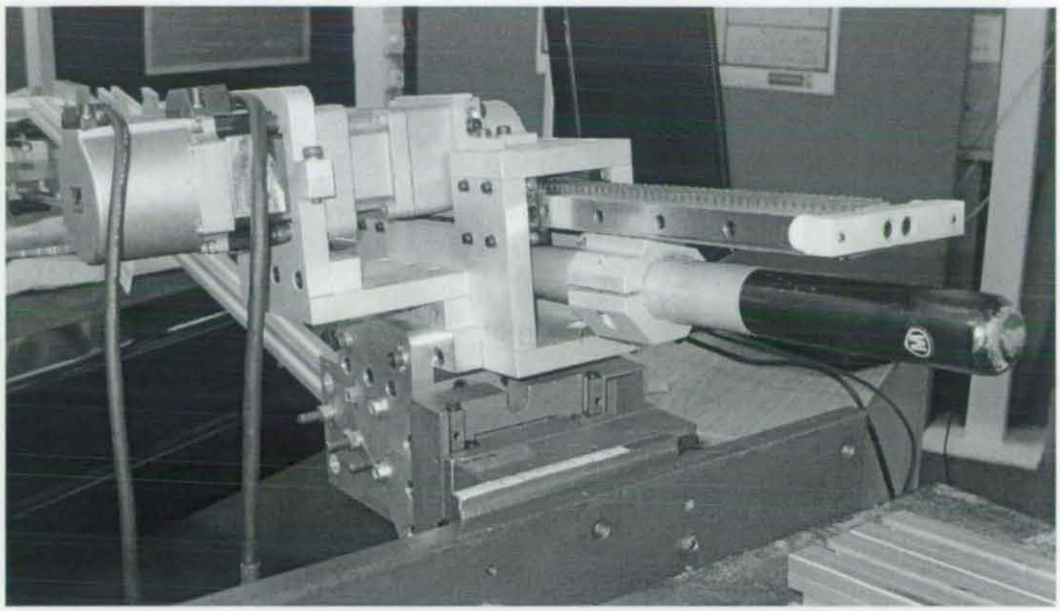
Typical Response Curve

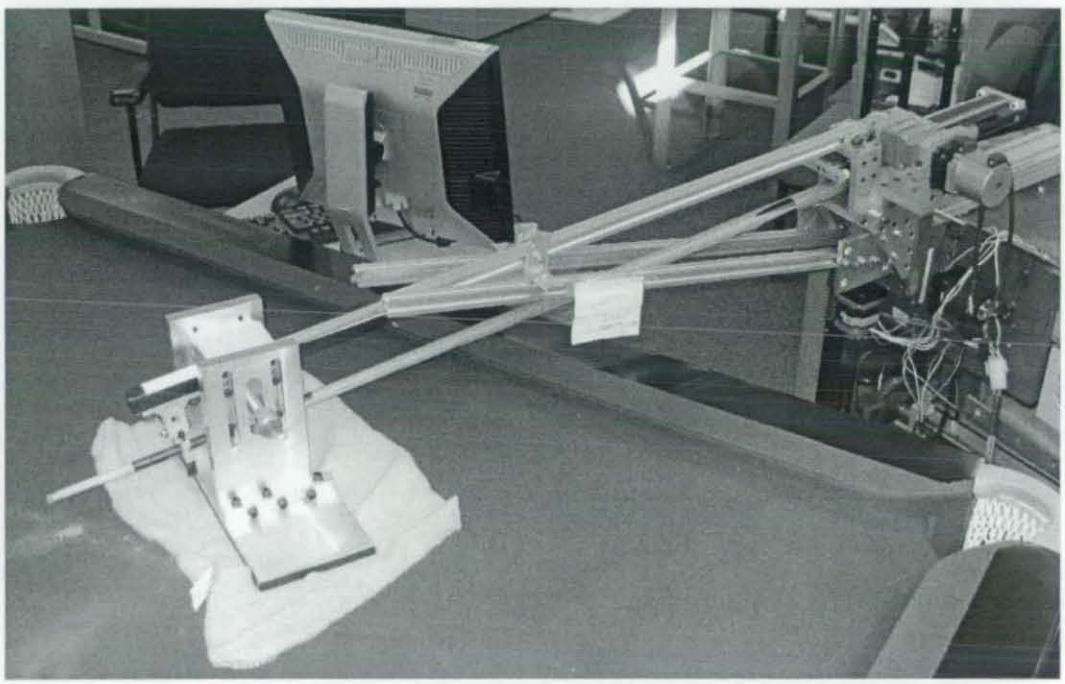


Some Pictures of the Robotic System





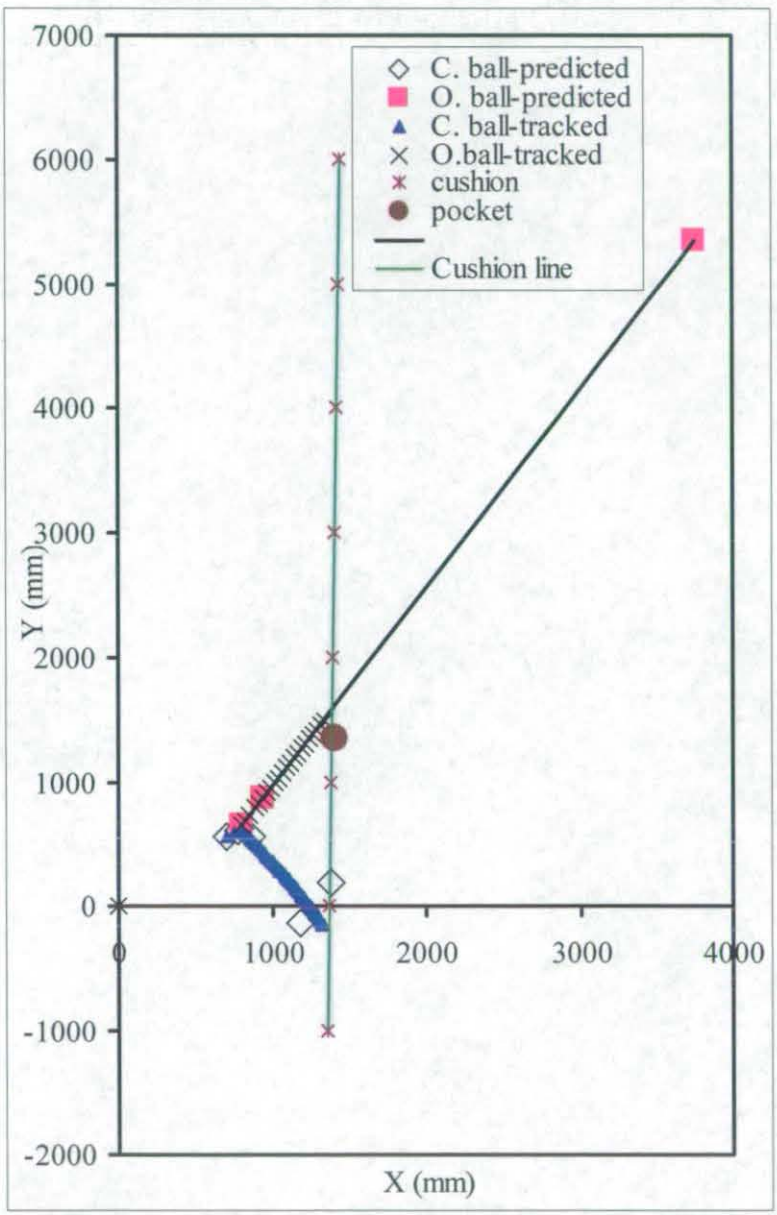




APPENDIX II: BALL POSITIONING RESULTS

Here some of the shots that were executed to test the validity of the forward dynamics model are given.

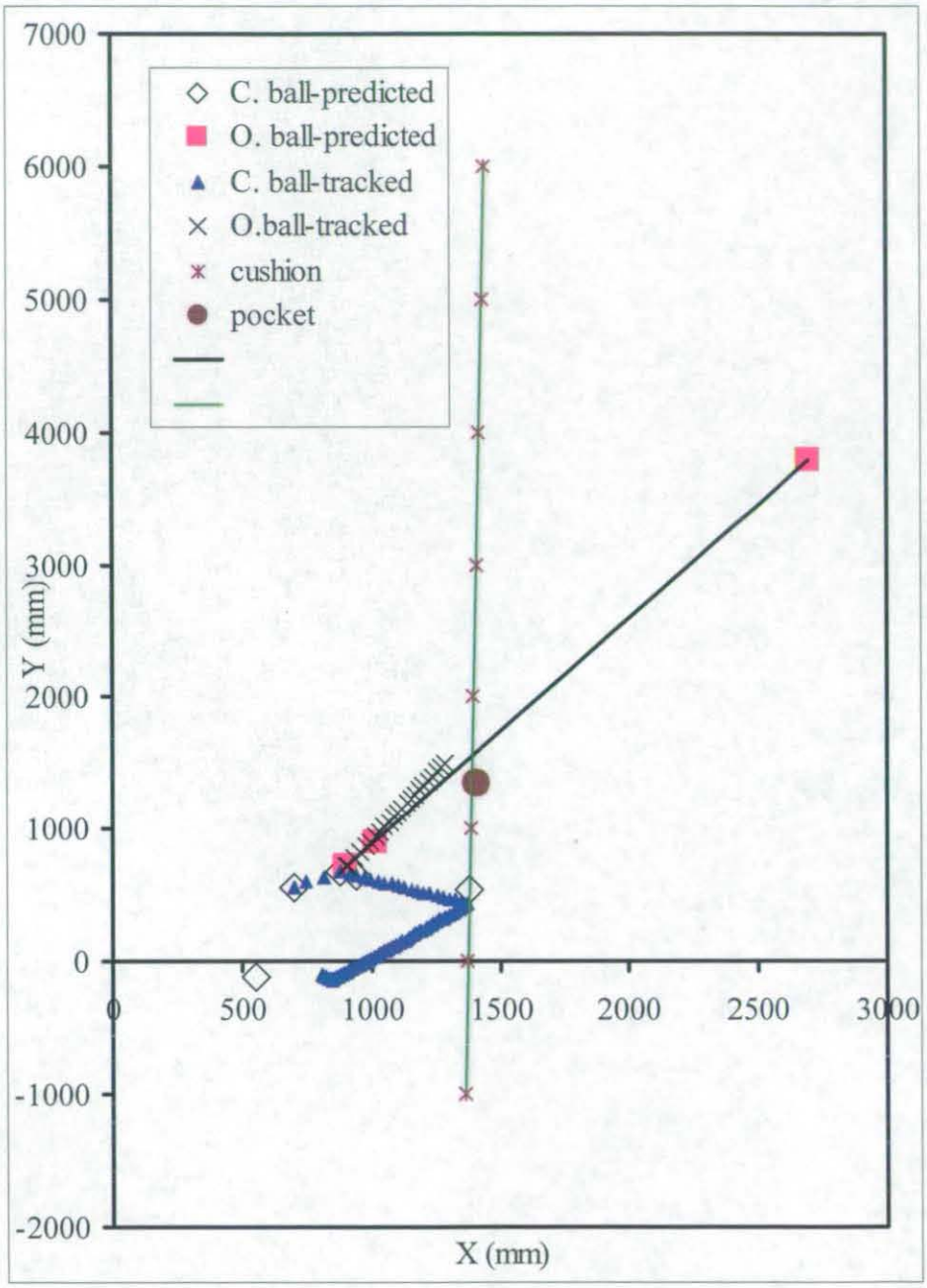
V_C: '100p' , x'_0 = 8 mm, y'_0 = -6 mm, θ_C = 29.5°.



Cue ball positioning error: 83 mm. A cue ball-cushion collision is predicted by the dynamics model, whereas no such collision occurred in the shot.

Error in the object ball path (estimated at the cushion intersection of the predicted path): 12 mm.

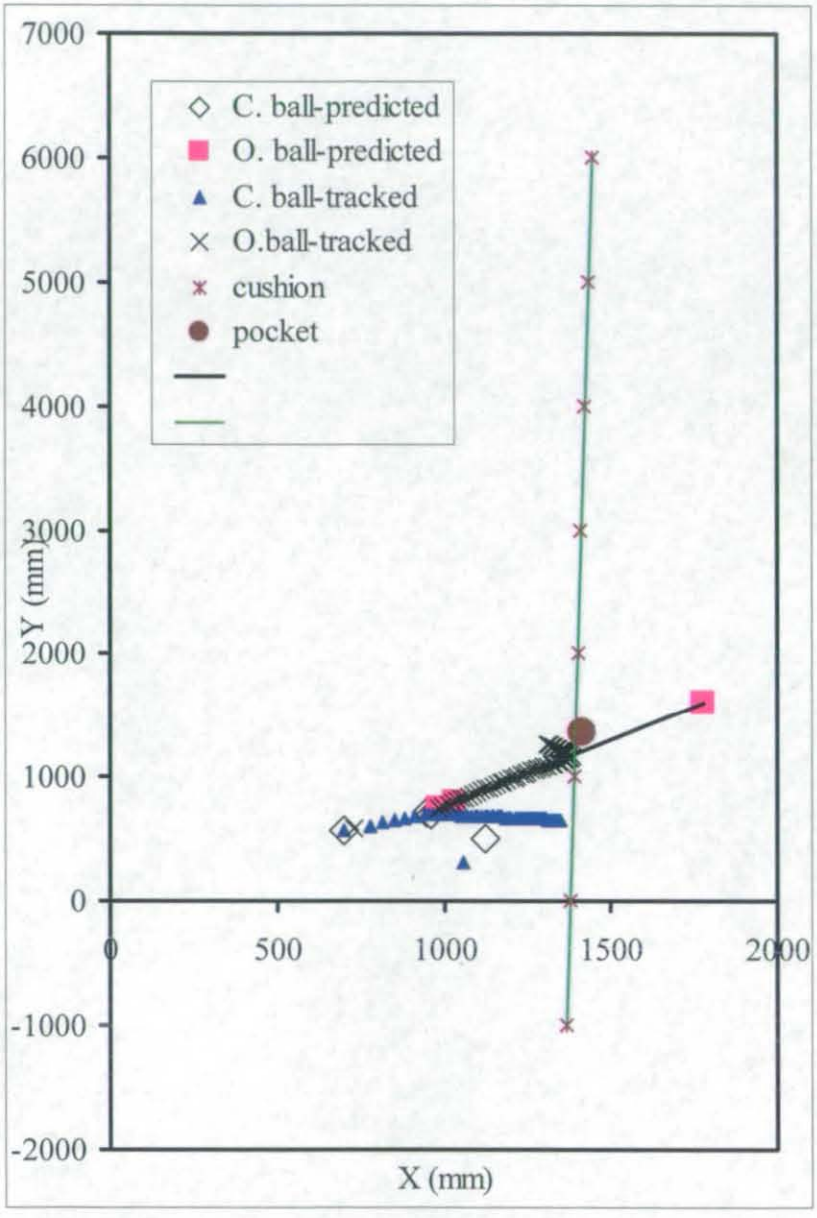
V_C: '100p' , $x'_0 = 0$ mm, $y'_0 = -10$ mm, $\theta_C = 29.5°$.



Cue ball positioning error: 248 mm. A cue ball-cushion collision is predicted by the dynamics model, whereas no such collision occurred in the shot.

Error in the object ball path (estimated at the cushion intersection of the predicted path): 100 mm.

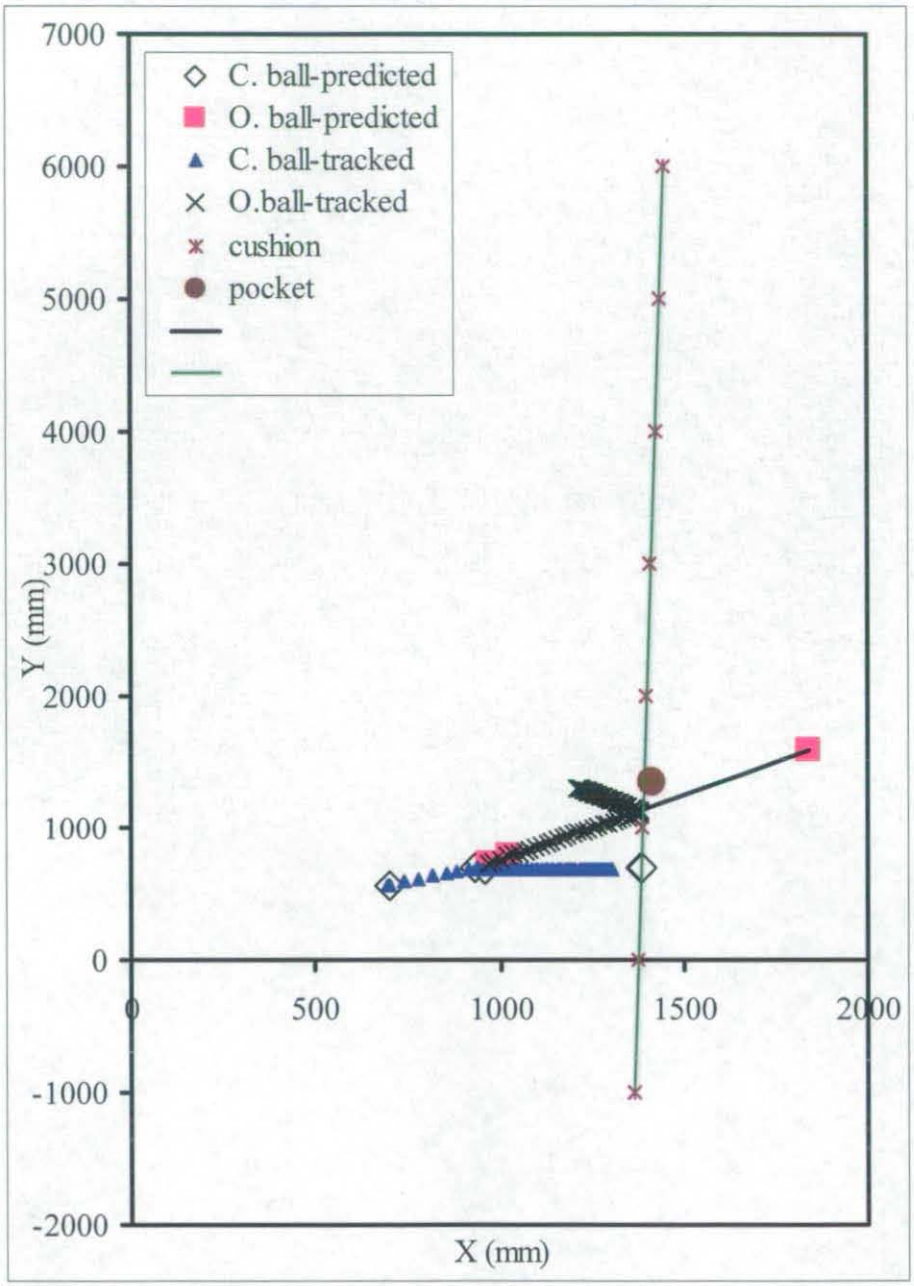
V_C: '140p' , $x'_0 = 0$ mm, $y'_0 = -10$ mm, $\theta_C = 29.5°$.



Cue ball positioning error: 303 mm. A cue ball-cushion collision is predicted by the dynamics model, whereas no such collision occurred in the shot.

Error in the object ball path (estimated at the cushion intersection of the predicted path): 29 mm.

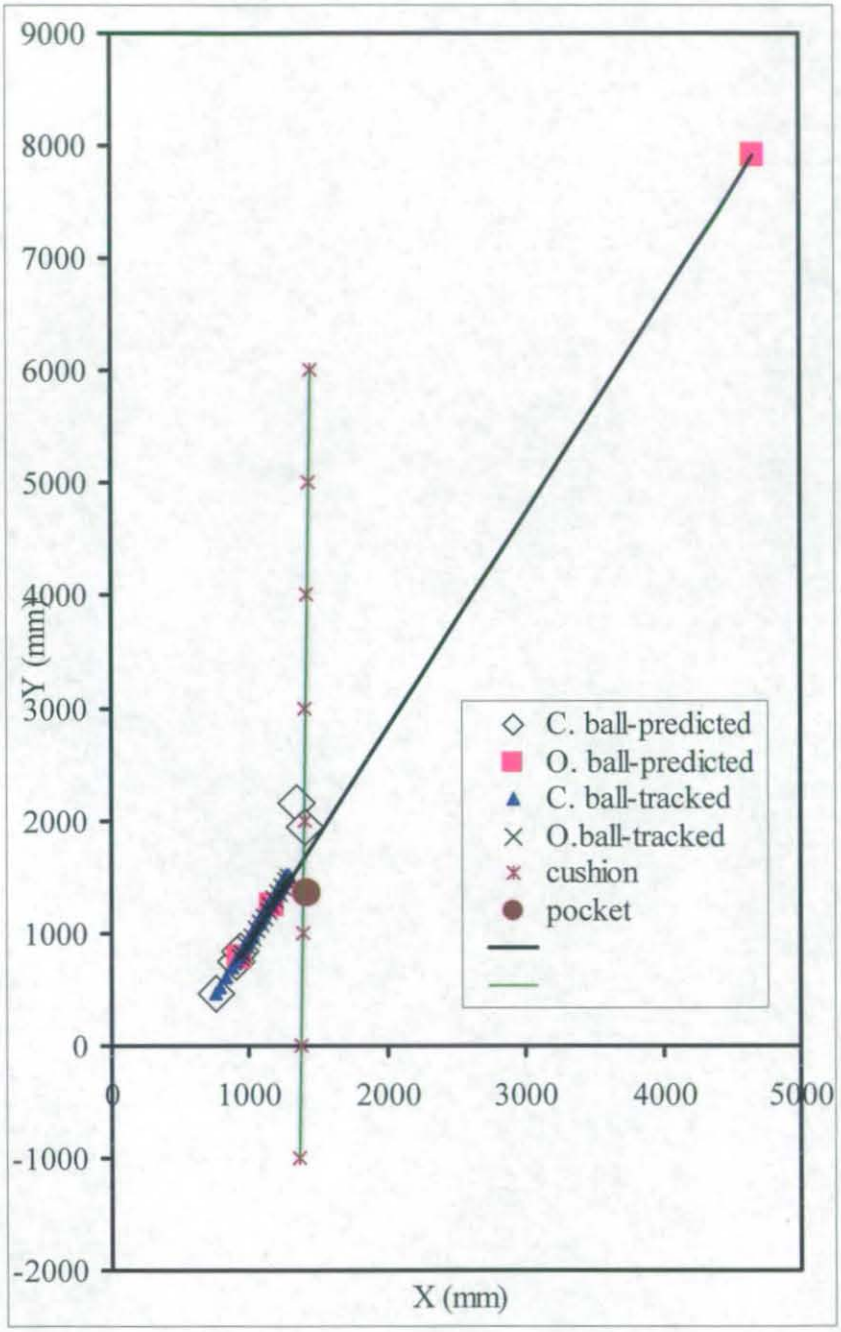
V_C: '160p' , $x'_0 = 0$ mm, $y'_0 = 0$ mm, $\theta_C = 29.5°$.



Cue ball positioning error: 74 mm. A cue ball-cushion collision is predicted by the dynamics model, whereas no such collision occurred in the shot.

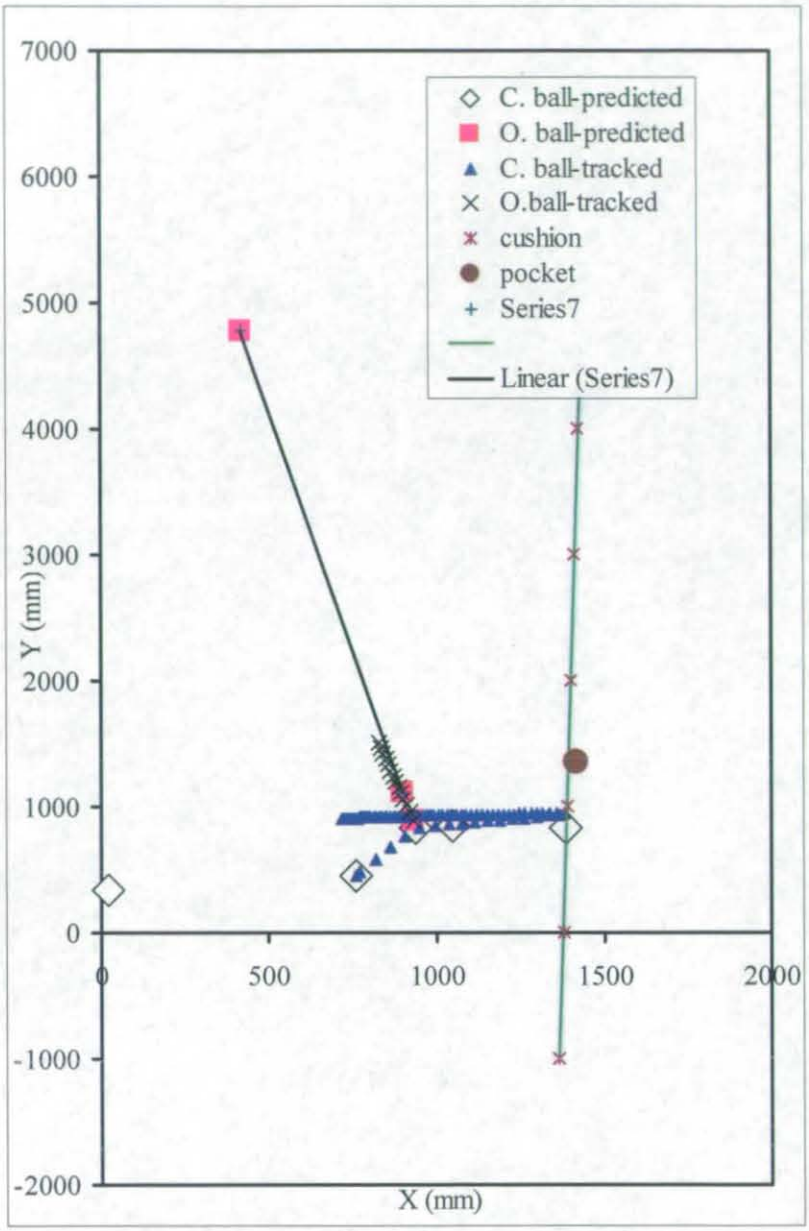
Error in the object ball path (estimated at the cushion intersection of the predicted path): 22 mm.

V_C: '100p' , $x'_0 = 0$ mm, $y'_0 = 10$ mm, $\theta_C = 63.8°$.



Cue ball positioning error could not be obtained as top spin was imparted to the cue ball in this case, and this took the cue ball further away at the direction of the object ball and eventually disappeared from the field of view. However the angle at which the cue ball was travelling coincided with the one that was predicted by the model indicating that it could end somewhere close to the predicted position. Error in the object ball path (estimated at the cushion intersection of the predicted path): 15 mm.

V_C: '100p' , $x'_0 = 6$ mm, $y'_0 = -6$ mm, $\theta_C = 63.8°$.



Cue ball positioning error: 930mm.

Object ball deviation (estimated at a distance of 2 m from the ball-ball collision location): around 160 mm. The object ball will not be potted for the distance of 2m from the collision location..

Such a large variation in both the positioning parameters suggests that a mishit may have happened due to jerk-induced vibration in the system, this especially happens whenever the front part of the robot was not properly clamped to the cue rest that sits on the table.

APPENDIX III: JOURNAL PAPER

Mathavan, S., Jackson, M.R., and Parkin, R.M (2009). Application of high-speed imaging to determining the dynamics of billiards. American Journal of Physics, Volume 77, Issue 9. pp: 788-794.

Application of high-speed imaging to determine the dynamics of billiards

S. Mathavan,[a] M. R. Jackson,[b] and R. M. Parkin[c]
Mechatronics Research Group, Wolfson School of Mechanical and Manufacturing Engineering, Loughborough University, Loughborough LE11 3UZ, United Kingdom

(Received 19 December 2008; accepted 2 June 2009)

In spite of interest in the dynamics of the billiards family of games (for example, pool and snooker), experiments using present-day inexpensive and easily accessible cameras have not been reported. We use a single high-speed camera and image processing techniques to track the trajectory of snooker balls to 1 mm accuracy. Successive ball positions are used to measure the dynamical parameters involved in snooker. Values for the rolling and the sliding coefficients of friction were found. The cushion-ball impact was studied for impacts perpendicular to the cushion. The separation angles and separation velocities after an oblique collision were measured and compared with predicted values. Our measurement technique is a simple, reliable, fast, and nonintrusive method, which can be used to test the numerous theories for the dynamics of billiards. The addition of a spin tracking element would further broaden its capabilities. © *2009 American Association of Physics Teachers.*
[DOI: 10.1119/1.3157159]

I. INTRODUCTION

Pool and snooker are popular billiard games. Billiard games involve very subtle physics and have been of interest to the physics community for over 200 years. The first extensive treatment of billiards was by Coriolis in 1835.[1] Other works, such as those of Wallace and Schroeder,[2] Salazar and Sanchez-Lavega,[3] and de la Torre Juárez,[4] address the dynamics of billiards. There have been both theory and experimental works[5] on the dynamics.

Special apparatus have been used for the measurements in most instances. For example, glass and textured black formica was used to replace the table-felt in studying the collisions between billiard balls.[2,5] Tracking techniques such as spreading talcum powder on the surface of the table have also been employed. Many of these techniques affect the dynamics that is being studied. Although Bayes and Scott[5] used a Polaroid camera and a stroboscope to track the balls, they did not base their results on this setup probably due to the poor accuracy of the cameras in the 1960s. As recent as 1994, rudimentary techniques were still used to estimate the physical parameters in billiard dynamics. For example, Marlow[6] used a meter stick and a stop watch to measure friction coefficients.

Today's technologies allow the high resolution tracking of objects. High-speed tracking technologies are extensively used in sports such as football, tennis, and cricket.[7,8] Alciatore[9] used high-speed video capture to visualize the dynamics in the game of pool. Alciatore[10] also used infrared imaging to visualize the collision points. However, he did not analyze the images to extract the physical parameters involved in the dynamics. Cross[11] employed a video camera to measure the ball velocity and ball spin using an overhead camera and analyzed squirt dynamics in a cue ball suspended as a pendulum bob. Researchers involved in robotic billiards have also used overhead cameras to locate the static ball positions on the table.[12–15]

In this paper we use high-speed camera based tracking to measure the characteristics of the interactions between the cue ball, table, and object ball. Accurate spatial and temporal tracking of the ball and the use of speed-time plot of the balls allow us to distinguish the different phases of ball dynamics, such as sliding, rolling, and impulses. The accurate detection of the changes in the phases of the ball motion allows us to measure the parameters more accurately than has been done. The use of speed-time plots also allows us to measure the effects of special collision between two balls, such as "overspinning," which has only been qualitatively described in the literature.

II. EXPERIMENTAL SETUP

A Riley Renaissance type snooker table with dimensions of 10×5 ft^2 was installed in our laboratory (see Fig. 1). This brand is the official table of the World Snooker Association and has been used for its professional snooker tournaments since 1992.

The tables used in pool and snooker are almost identical, except that the pool table has larger pockets compared to the size of a pool ball. At the start of a game there are 21 colored balls worth various points and a white cue ball at predefined places on the table.

A machine vision camera was mounted on the ceiling, right above the snooker table, looking vertically downward (Fig. 1). A single camera is sufficient to capture the dynamics because the dynamics is confined to the table surface. The color camera is PixeLINK PL-B776F with 3.15×10^6 pixel resolution. The camera is connected to a host personal computer via FIREWIRE. For the region of interest option the camera is capable of capturing up to 1000 frames per second (fps). This feature of the camera was used whenever it was necessary to analyze the dynamics at finer temporal resolutions. The camera is fitted with a wide-angle lens to capture the whole table from the limited available headspace between the snooker table and the ceiling. The table area is imaged to a 1 mm spatial resolution with the current setup of the camera.

To verify that the measurements made by the camera are accurate, some distance measurements were also made with a meter stick. For this purpose two rectangular blocks with a height of the ball radius and with circular white patterns on their top surfaces were placed at two locations on the table. Circular patterns of diameter of 52.4 mm (the ball diameter) were used so that the camera and the image processing algorithm would treat them as balls. The distance between their centers was obtained using the camera and the meter stick.

(a) (b)

Fig. 1. Snooker table and ceiling-mounted machine vision camera in the mechatronics laboratory. Note the headspace and the vertical mount of the camera to look perpendicularly down at the table.

We used this method because it was very cumbersome to physically measure the center distance between two snooker balls because the balls change position with the slightest touch. This procedure was repeated for several random positions of the blocks almost covering the whole imaged area of the table. The differences in the measurements by the two methods were found to be at most 2 mm, validating the results from the imaging system. The video and image handling and the image processing were performed using MATLAB.

A. Methods

Before measurements could be made on the images from a camera, two calibrations were done. The intrinsic camera calibration was performed to correct for the lens distortion that is present in wide-angle lenses [see Fig. 2(a)]. The camera calibration toolbox from the Computational Vision Group at Caltech was used in conjunction with MATLAB to calibrate the camera; for a detailed description of the procedure, see Ref. 16.

The MATLAB toolbox also incorporates an extrinsic calibration element. The extrinsic calibration procedure enables metric measurements to be made from the values given in terms of pixels. This procedure provides the translation and rotation matrices that relate the real world coordinate system to the image plane (see Fig. 3). The equation for the transformation between a point in the world frame xyz to its corresponding image point in the camera fame $x'y'z'$ is $x' = Rc^*x + Tc$, where Rc and Tc are the rotation and translation matrices, respectively.[17]

A real world coordinate system was selected such that it was fixed to the snooker table so that two of its axes lie along the two perpendicular edges of the table and both x and y lie on the imaginary plane that is created by the ball centers [see Fig. 2(b)], which is 26.2 mm above the table surface. The

(a) (b)

Fig. 2. (a) Distorted and (b) corrected images of the half table (note the barrel distortion due to the wide-angle lens) with the checkerboard pattern, for extrinsic camera calibration, placed on the table.

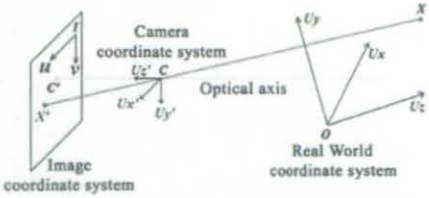


Fig. 3. The pinhole camera model shows how a real world point X is projected as X' on the camera image plane uv, through the optical center C of the lens. Also note how the camera frame $x'y'z'$ is fixed to the optical center of the lens.

experiments were performed in half of the table area to obtain better spatial resolution from the camera. The image blur due to fast moving balls was minimized by selecting the lowest possible camera shutter opening time. Image sequences with high image blurs were not analyzed. Quantification was done by counting the number of pixels in a blur and then comparing it with the number of pixels found in a stationary ball.

B. Image processing

An image processing program was written to execute the following operations. The video was captured and then split into image frames. The images were then converted into gray scale images. Each of these gray scale images were then transformed to binary images using an appropriate threshold value of the image intensity. A treatment of these concepts can be found in textbooks on digital image processing such as that of Gonzalez and Woods.[18] Then the image processing program to extract the ball centroid was executed. Two functions from the MATLAB Image Processing Toolbox called *bwlabel* and *regionprops* were used to extract the ball from the image, thus determining its centroid in pixels. The real world coordinates of the ball centroid are obtained using the transformation matrices Rc and Tc from the extrinsic calibration procedure. The time stamping of these values based on the camera frame rate enables us to calculate the velocities and accelerations of the ball.

III. RESULTS AND DISCUSSION

The tracked cue ball is shown with its initial position on the snooker table in Fig. 4. The spatial separation between the successive tracked centroids indicates the variation in the ball's velocity.

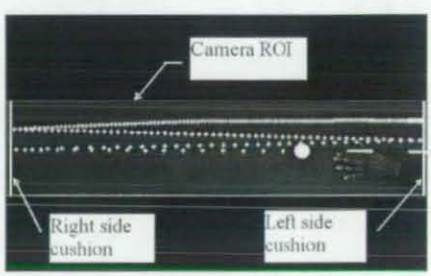


Fig. 4. The tracked cue ball positions (the centroid of the ball is shown by the white markers) are superimposed on the image at the start of the tracking, also showing the initial cue ball location (four consecutive impacts with two parallel cushions are shown).

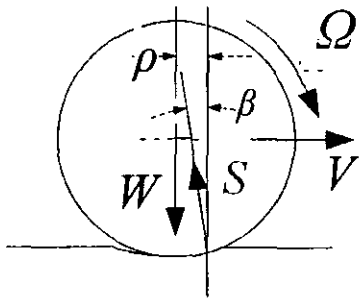


Fig. 5. The ball rolling on the table. It shows the forces that are acting on the ball while it is rolling. Note the reaction force from the table S, which is a combined effect of the "regular" normal reaction from the table and, most importantly, due to of the table-felt under the weight of the moving ball. The horizontal component of S, $S \sin \beta$, decelerates the ball.

A. Ball motion against surface friction on the table

When there is no relative velocity between the ball and the table at their contact point, the ball is said to roll on the table. During the rolling the linear and angular velocities of the ball, V and Ω, respectively, satisfy the relation $V=R\Omega$, with R the ball radius. Because the table-felt is deformable and the ball is rigid, the table surface deforms when the ball is in motion as shown in Fig. 5. Hence the ball makes contact with the table over an extended area. According to Ref. 19 this deformation is independent of V. The table cloth deformation results in a normal reaction force S from the table at an angle β with the vertical, inclined from the moving direction of the ball as shown in Fig. 5. For an extensive treatment of this deformation, see Refs. 19 and 20.

According to Fig. 5, the reaction force S has a horizontal component equal to $S \sin \beta$, which opposes the ball motion. Generally the reaction force S does not go through the centroid of the ball, and hence there is a torque acting in the opposite direction to that of the angular velocity Ω, resulting in angular deceleration. The rolling friction coefficient does not change with the ball's velocity and is a constant because it depends only on the surface properties of the table-felt and the geometry and mass of the ball.[19]

Figure 6 shows the variation in the ball's velocity with respect to time. Once the impulse is delivered to the ball, the ball's velocity decreases rapidly during what is known as the sliding phase, and then the ball starts to roll. Reference 3 showed that the ball starts to roll immediately only when the ball is hit horizontally at a height of $7R/5$ from the table surface. In Fig. 6 the velocity gradient during the rolling phase gives the value of the deceleration due to rolling friction. Different shots were tracked and the deceleration during the rolling phase was found to be 0.124–0.126 m/s^2. The rolling friction coefficient, which is usually expressed as a fraction of the gravitational acceleration 9.81 m/s^2, is 0.0127–0.0129. Marlow[6] suggested a range of 0.011–0.024 for pocket billiards (pool) and a mean value of 0.016. Although the physical properties of the ball and table are different in pool and snooker, there is no obvious reason for this excessive variation (more than 100% of the lower value) obtained in pool with Marlow's measurements. The only plausible explanation is that the meter stick and stop watch measurement method used by Marlow is prone to error. Although Williams[21] claimed that the nap of the table-felt affects the ball motion, depending on whether its motion is toward the top cushion or away from it, we did not find any evidence to support this claim.

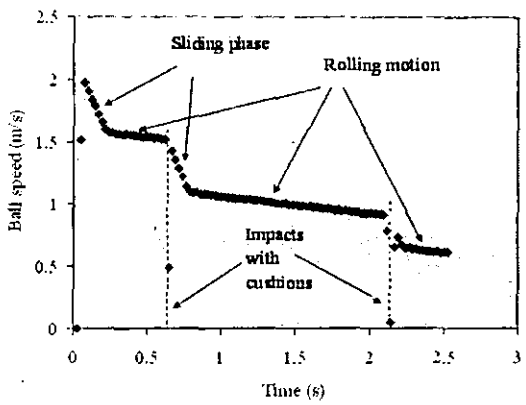


Fig. 6. The speed-time plot for the ball showing all the different phenomena involved from the video captured at 42 fps (the complete motion profile until the ball comes to the rest is not shown here).

When there is a relative velocity between the ball and the table at their point of contact, the ball is said to slip on the table. In the sliding phase $V=R\Omega$. For a theoretical treatment of all the possible cases of ball motion immediately after the cue impact, see Ref. 3. The friction that exists during the sliding motion (the sliding coefficient of friction) usually depends on the sliding velocity of the ball. The ball speed-time plot given in Fig. 6 shows that the sliding friction is much larger than the rolling friction, disappears within a very short time interval, and quickly diminishes with the velocity. Another interesting observation from this plot is that after the ball has started its rolling motion, it starts to slide again (note the speed gradients immediately after the impacts) when it collides with the cushion (table wall/rail) because the cushion impact violates the $V=R\Omega$ rolling condition. Once $V=R\Omega$ is reached again, the ball goes into the pure rolling mode.

From the analysis of the speed of the tracked ball, the sliding friction coefficient was found to be in the range of 1.75–2.40 m/s^2 (0.178–0.245 in dimensionless units). These values were obtained for the ball motion along random directions on the table. Marlow[6] calculated a dimensionless value of 0.2 for pool using the rolling coefficient value of 0.016. An independent measurement was not performed because only a meter stick and a stop watch were available. Witters and Duymelinck[22] used stroboscopic illumination to photograph a decelerating pool (not snooker) ball. They found that when the ball velocity increases from zero, the friction coefficient approaches 0.21 from a value of 0.14. Such a variation could not be verified from our experiments.

The sliding friction is 15–20 times larger than the rolling friction. Also, during the sliding phase some rolling action will simultaneously take place, as the displacement effect, shown in Fig. 5, is always present at the ball-table interface. Due to its comparatively small magnitude (approximately 5%), it is usually neglected, and the motion is treated as pure sliding.

B. Ball-cushion interaction

To visualize and analyze the impulse dynamics between the ball and the cushion, high-speed image capturing experi-

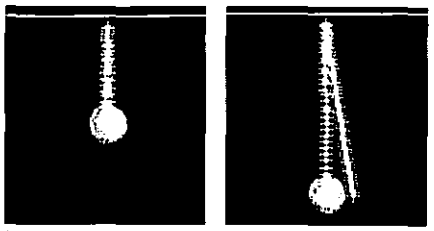


Fig. 7. Bounce of the cue ball off the rail. The ball location depicts its position as it approached the rail. The frame rate is 120 fps, and an imaginary continuous white line shows the approximate location of the cushion.

ments with >100 fps for small regions of interest were performed. The impulse of the cue ball on the cushion depends on factors such as the speed at which it collides with the cushion surface, the incident angle with respect to the cushion surface, the amount of spin of the ball, the physical characteristics of the ball and the cushion, and the parameters involved in the interaction between the ball and the cushion such as the coefficient of restitution and the surface friction.

Spin on the ball changes the impact characteristics drastically. Ball spin is difficult to quantify with our experimental setup and methodology. Sidespin changes the postimpulse cue ball path significantly; the interested reader is directed to Ref. 23 or Ref. 9. The ball-cushion interaction is a case of multiple impacts, both normal and tangential, the latter due to the force of friction, with one component normal to the cushion surface, and the other two perpendicular frictional impacts from the cushion wall. Derivations of the dynamics for general impact are not available.

For this reason we conducted experiments on shots without considerable sidespin. Care was taken so that a shot was directed perpendicular to the rails (cushions) as much as possible. Whenever the cue ball is played perpendicular to the rails, if it does not have any sidespin and should bounce back along the same path along which it approached the rail. This criterion was used to ensure that the shots did not impart a considerable sidespin on the cue ball. Figure 7(a) shows a perpendicular shot with no sidespin, and Fig. 7(b) shows a perpendicular incoming shot that apparently has some sidespin, which results in the ball rebounding to the right side. For the rebound analysis the shot shown in Fig. 7(b) was not used, and only the one shown in Fig. 7(a) was used. The no-sidespin condition ensures that there is only one unknown in the form of top/back spin.

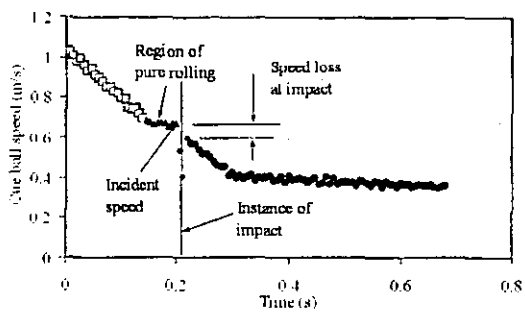


Fig. 8. The tracked results for a ball-cushion impulse (at 150 fps).

If we assume that the ball had gone into pure rolling mode before the impact, we can determine the top spin of the ball from $\Omega = V/R$. Thus the incident ball velocity V is the only independent variable involved, and the velocity drop during the impact can be correlated with V.

Figure 8 shows the velocity plot obtained from a high-speed video captured at 150 fps. The velocity plot was used to determine if the ball was rolling just before it hit the cushion. The gradient of the speed-time plot was used to determine this, as shown in Fig. 8. Results that were obtained for 31 such shots into the rails, almost satisfying the conditions of no-sidespin and that of pure rolling, are given in Fig. 9.

From Fig. 9 we see that the relation between the rebound and incident speeds is almost linear for the incident velocity in the range of 0.28–3.5 m/s (the typical range of ball velocities in the game). A best fit straight line for the rebound-incident speed data gives a coefficient of restitution of 0.818 for this velocity range. The results are more closely fit by the second-order polynomial $y = -0.0877x^2 + 1.131x - 0.0953$, where x is the incident velocity and y is the rebound velocity. These results are not valid for a general ball-cushion impulse but are applicable only under the conditions of no-sidespin and pure rolling motion prior to the impulse. We believe that the ideal variation between the rebound and incident speeds should be linear and the reduction in the coefficient of restitution at higher incident speeds is due to cushion deformation. The gradient of the plot at lower incident speeds is around 0.910, and this value shall be valid under the assumption of a rigid cushion.

Marlow[6] reported that the coefficient of restitution for rails in a billiard table is 0.55 but did not give much detail about the experimental procedure. He compared his results with the values suggested by Coriolis[1] and concluded that they agree closely.[6] The cushion height for snooker is 36 mm, with the ball radius equal to 26 mm, which is close to the height of 1.4 times ball radius found in pool. Thus the cushion and ball geometry is almost identical in pool and snooker. It is possible that Marlow considered the rebound ball velocity at the end of the sliding phase rather than the correct one immediately after the impulse. Then the coefficient of restitution for the shot could be 0.63, but this result has no physical meaning.

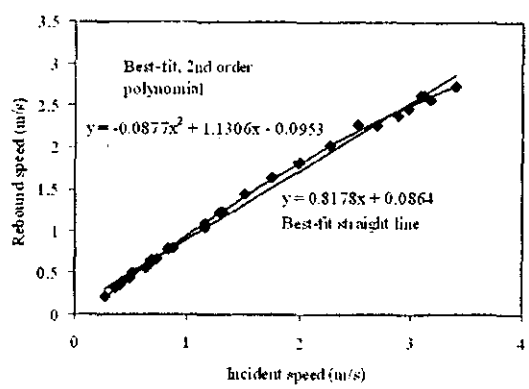


Fig. 9. The variation in the rebound velocity versus the incidence velocity. At lower incident velocities the variation is almost linear. However, at higher incident velocities the rebound velocities tend to level off, quite possibly as the cushion is not rigid at higher incident speeds.

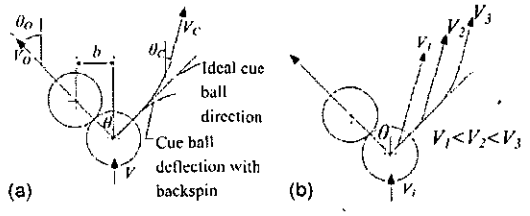


Fig. 10. The effect of table friction on the cue ball path for an oblique collision (from Ref. 2). (a) The parameters involved in an oblique collision. (b) The cue ball path for different precollision cue ball speeds under rolling conditions.

C. Impact between balls

If the approaching and separating velocities of two balls lie along the line connecting the centers of the balls, then the impact is said to be frontal or head-on. Impacts occur in two dimensions in billiards as oblique collisions, and the frontal impact is a special case.

Amateur billiards players use the 90° rule[9,10] to visualize the postcollision trajectories of the colliding balls. It states that the balls will separate at 90° after an oblique collision (see Fig. 10 for the predicted ideal directions of travels). It is also assumed that the cue ball will immediately stop after a frontal collision. In snooker the cue ball and all object balls have the same mass. It can be easily shown by momentum conservation that the 90° rule only holds when the coefficient of restitution between the balls is one (that is, the balls are purely elastic). The angular velocity of the cue ball (in the form of the side/top spin) when it collides with the object ball also affects the postcollision velocities and the directions of separation for the balls. The friction present between the colliding balls has also been shown to affect the postcollision motion.[24] Bayes and Scott[5] employed a spring loaded cue launcher and two pool balls on a felt-covered table to examine this effect. They used a stroboscope and a camera to determine the subsequent ball paths and found that the angle was around 67°. There is no data on how much spin the ball had at the time of impact, which is known to affect the collision dynamics. They also tested the ball on various glass surfaces and found that the collision angle approaches 90° as the surface becomes smoother (in soapy glass it reached 89.9°). Thus table friction creates some unpredictable behavior in the ball collision.

The tracked results for the cue ball and an object ball collision are shown in Fig. 11. We see that the temporal resolution of the tracking is sufficient to capture the deflections in its postimpact trajectory. The reason for the curvature in the path of the cue ball is that it starts to slip immediately after the impact [a similar slipping phenomenon was also observed in the ball collision with a cushion; see Fig. 10(a)]. Figure 10(b) gives an idea of how this behavior is influenced by the incident velocity of the cue ball. Similarly, the object ball also starts to slip immediately after the impact. Once the slipping phase has stopped, both balls go into rolling motion, and the curved path of the cue ball is then directed along the tangent line to the curve. Reference 2 analyzed this phenomenon and showed that the velocities for the postcollision and postsliding phases of the object ball are (see their notation in Fig. 10)

$$V_O = \tfrac{5}{7} V \cos\theta,$$

$$\theta_O = \theta, \tag{1}$$

and for the cue ball is

$$V_C = \tfrac{5}{7} V \sqrt{\tfrac{9}{5}\sin^2\theta + \tfrac{4}{25}},$$

$$\theta_C = \tan^{-1}\left[\frac{\sin\theta\cos\theta}{\left(\sin^2\theta + \frac{2}{5}\right)}\right]. \tag{2}$$

They defined $\beta = b/D$ as the fractional impact parameter, where D is the ball diameter and b is the separation of the ball centers in the direction perpendicular to the incident ball velocity V. Also note that $\beta = \sin\theta$.

Plots of angles θ_o, θ_c, and $\theta_o + \theta_c$ versus the impact parameter are shown in Fig. 12. The experimental values agree with the theoretical predictions in most instances, but θ_o deviates more from its theoretical value at high fractional impact values. The reason is unknown, and we do not know if factors such as spin affect collisions for very oblique collisions. A possible explanation is that at high values of b, an excessive amount of sidespin is imparted to the object ball, which changes its path from what is derived in Ref. 2. This phenomenon also raises questions about whether sidespin affects its speed or direction of travel.

In billiards sidespin is considered to be independent of the linear speed of the ball because it is assumed that the ball makes a point contact with the table. If both the contacting surfaces are extremely rigid, this assumption would be valid. For billiards the rigid table-top is covered by a soft felt. Thus a considerably rigid billiard ball sinks into the felt, making contact over a finite region of the ball's surface. Hence we suspect that the ball exhibits disklike properties. For a flat disk, such as an ice-hockey puck, its linear motion and its

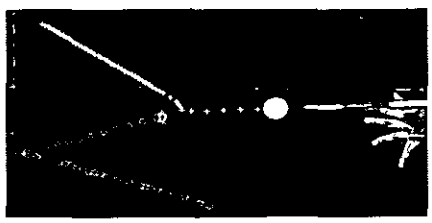


Fig. 11. Tracking results for a collision between the cue ball and an object ball at 45 fps.

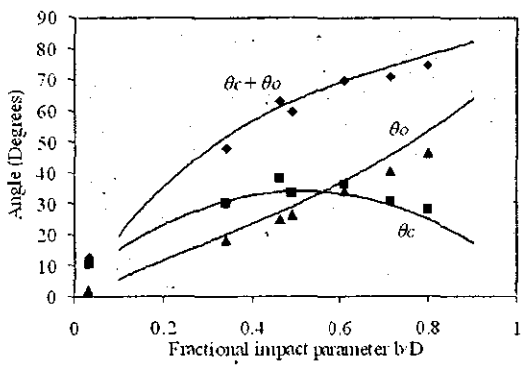


Fig. 12. Theoretical and measured deflection angles for the cue and object balls versus the fractional impact parameter β. The symbols ■, ▲, and ◆ represent θ_c, θ_o, and $\theta_c + \theta_o$, respectively. Continuous lines show the respective predictions.

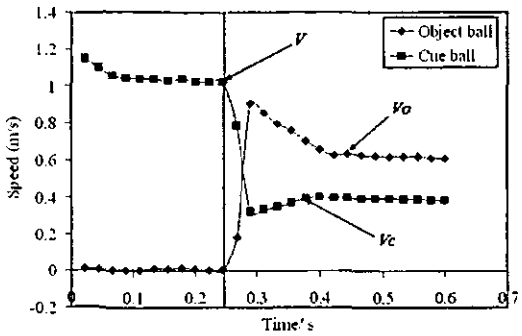


Fig. 13. A typical speed variation in the cue ball and object ball near impact. It shows how both the cue ball and the object ball start to slip on the table immediately after their collision. The cue ball speed plot also shows how the cue ball is accelerated after the impact.

rotation (it only has a side-rotation, which is analogous to the sidespin of the ball) are always coupled.[25] That is, the rotational motion and linear motion of a disk will end at exactly the same time.[25] Thus there are some characteristics of the disklike motion found in billiard ball motion. One observation that supports this claim is that we never see the ball continue to rotate about the vertical axis (that is, sidespin) after its linear motion is stopped. The coupling of linear and rotational motions is readily apparent in the game of pool where the balls have a number or other pattern painted on their surface.

Wallace and Schroeder[2] did not experimentally validate the velocity relations found in Eqs. (1) and (2) because their tracking method could record only the positions and not the time stampings. We use the velocity plots for both colliding balls to validate these equations. As shown in Fig. 13, the incident velocity is measured right at impact. The gradient, typical for the pure rolling motion, as discussed, was used as the criteria for detecting the time at which the ball starts to roll (or stops slipping). The detected times are shown with their respective velocity symbols in Fig. 13. We observe that the cue ball accelerates right after the collision. This acceleration occurs because the collision greatly reduces only the linear velocity and not the angular velocity, and thus the cue ball goes into a sliding condition with excess top spin (overspinning). This excess top spin is then converted into linear velocity by the action of the sliding force, which in this case acts in the same direction as the ball velocity, increasing the latter.

The results are given in Table I for five such shots involving impacts. The maximum error between the theory and the measurements is found to be around 10%. We do not know whether this error is also induced by the effect of sidespin on the collision between two balls. The sidespin of the ball was not taken into consideration in Ref. 2. There is reason to believe that the friction between the cue ball and the object ball will introduce tangential force components at the collision point, which would impart a sidespin onto the object ball, even though the values of these tangential force components may be small.

During impulse there will be a relative velocity between the cue ball and object ball along the vertical due to the angular velocity (top spin) in the natural roll of the cue ball prior to the impact. This relative velocity will introduce a tangential friction force during the time of impulse on the cue ball as well as on the object ball. This force will induce a spin on the cue ball about its frontal velocity axis, producing an effect equivalent to a massé shot (a shot played with an elevated cue stick). For a massé shot a ball is known to move along a curved path instead of on a straight line. This correction should also be added to the prediction in Ref. 2. These observations and the evidence presented in Fig. 12 and Table I should motivate a new theory for the collision between two balls, which involves the frictional forces between the balls that are present during the impulse.

IV. CONCLUSIONS

High-speed video capture using a single machine vision camera was found to give good results in determining the dynamics involved in snooker. The rolling coefficient of friction was found to be between 0.124 and 0.126 m/s^2. The sliding friction value is in the range of 1.75–2.40 m/s^2. One-dimensional ball-cushion collisions were also analyzed, and the mean coefficient of restitution was determined. Both frontal and oblique collisions between the balls were analyzed. Predictions of separation angles and velocities were tested experimentally and close agreement was found.

Some experiments could not be performed. One such experiment would look at the general impact of the ball with the cushion. The inability to perform such experiments is mainly due to the difficulty of determining the amount of spin on the ball using the camera. To track the ball spin in football and golf, researchers have used marked patterns on the ball surface. Some interferometer based techniques have also been used for this purpose.

Table I. The postimpact speed theoretical predictions (Ref. 2) and the measured values from ball tracking. V is the incoming cue ball speed, θ is the cut angle for oblique collision, V_o and θ_o are the postcollision and postslip direction of travel and speed for the object ball, and V_c and θ_c are the postcollision and postslip direction of travel and speed of the cue ball.

V (m/s)	θ (°)	Measured V_c (m/s)	Measured V_o (m/s)	Theoretical V_c (m/s)	Theoretical V_o (m/s)	Error in V_c (%)	Error in V_o (%)
1.539	33.83	0.816	0.836	0.932	0.913	12.4	8.43
1.032	26.36	0.520	0.629	0.529	0.660	1.70	4.70
1.364	40.52	0.925	0.700	0.934	0.740	0.964	5.40
1.731	46.50	1.275	0.787	1.301	0.851	2.00	7.52
0.942	18.05	0.365	0.581	0.388	0.640	5.93	9.22

ACKNOWLEDGMENTS

The authors are indebted to the anonymous reviewers for their insightful, critical, and enthusiastic comments that resulted in an improved manuscript.

[a]Electronic mail: s.mathavan@lboro.ac.uk
[b]Electronic mail: m.r.jackson@lboro.ac.uk
[c]Electronic mail: r.m.parkin@lboro.ac.uk

[1]G.-G. Coriolis, *Théorie Mathémathique des Effets du Jeu de Billard* (Carilian-Goeury, Paris, 1835) translated by David Nadler [*Mathematical Theory of Spin, Friction, and Collision in the Game of Billiards* (David Nadler, USA, 2005)].
[2]R. E. Wallace and M. C. Schroeder, "Analysis of billiard ball collisions in two dimensions," Am. J. Phys. **56** (9), 815–819 (1988).
[3]A. Salazar and A. Sanchez-Lavega, "Motion of a ball on a rough horizontal surface after being struck by a tapering rod," Eur. J. Phys. **11**, 228–232 (1990).
[4]M. de la Torre Juárez, "The effect of impulsive forces on a system with friction: the example of the billiard game," Eur. J. Phys. **15** (4), 184–190 (1994).
[5]J. H. Bayes and W. T. Scott, "Billiard-ball collision experiment," Am. J. Phys. **3** (31), 197–200 (1962).
[6]W. C. Marlow, *The Physics of Pocket Billiards* (MAST, Palm Beach Gardens, FL, 1994).
[7]G. Pingali, A. Opalach, and Y. Jean, "Ball tracking and virtual replays for innovative tennis broadcasts," in Proceedings of the 15th International Conference on Pattern Recognition, Barcelona, Spain, 2000, Vol. 4, pp. 152–156.
[8]K. Davis, "A watchful eye" (www.matrox.com/imaging/news_events/feature/archives/2002/hawkeye.cfm).
[9]D. G. Alciatore, *The Illustrated Principles of Pool and Billiards* (Sterling, New York, 2004).
[10]D. G. Alciatore, "Pool and billiards physics principles by Coriolis and others" (http:billiards.colostate.edu/physics/Alciatore_AJP_MS22090_revised_pool_physics_article.pdf).
[11]R. Cross, "Cue and ball deflection (or "squirt") in billiards," Am. J. Phys. **76** (3), 205–212 (2008).
[12]F. Long, J. Herland, M.-C. Tessier, D. Naulls, A. Roth, G. Roth, and M. Greenspan, "Robotic pool: An experiment in automatic potting," in Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Sendai, Japan, 2004, pp. 2520–2525.
[13]A. W. Moore, D. J. Hill, and M. P. Johnson, "An empirical investigation of brute force to choose features, smoothers and function approximators," in *Computational Learning Theory and Natural Learning Systems 3: Selecting Good Models* (MIT, Cambridge , MA, 1995), pp. 361–379.
[14]S. W. Shu, "Automating skills using a robot snooker player," Ph.D. thesis, University of Bristol, 1994.
[15]M. E. Alian, S. E. Shouraki, M. T. M. Shalmani, P. Karimian, and P. Sabzmeydani, "Roboshark: A gantry pool player," 35th International Symposium on Robotics, Paris, France (2004) (www.cs.sfu.ca/~psabzmey/personal/publ/papers/alian_roboshark_isr04.pdf).
[16]J. Y. Bouguet, Camera calibration toolbox for MATLAB (www.vision.caltech.edu/bouguetj).
[17]J. Heikkilä and O. Silvén, "A four-step camera calibration procedure with implicit image correction," in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'97), San Juan, Puerto Rico, 1997, pp. 1106–1112.
[18]R. C. Gonzalez and R. E. Woods, *Digital Image Processing* (Prentice-Hall, Upper Saddle River, NJ, 2002).
[19]J. Hierrezuelo and C. Carnero, "Sliding and rolling: The physics of a rolling ball," Phys. Educ. **30** (3), 177–182 (1995).
[20]A. Domenech, T. Domenech, and J. Cebrian, "Introduction to the study of rolling friction," Am. J. Phys. **55** (3), 231–235 (1987).
[21]K. Williams, *Know the Game: Snooker*, 3rd ed. (A&C Black, London, 2002).
[22]J. Witters and D. Duymelinck, "Rolling and sliding resistive forces on balls moving on flat surface," Am. J. Phys. **54** (1), 80–83 (1986).
[23]J. Walker, "The physics of the follow, the draw and the massé (in billiards and pool)," Sci. Am. **249** (1), 124–129 (1983).
[24]A. Domenech and E. Casasus, "Frontal impact of rolling spheres," Phys. Educ. **26** (3), 186–189 (1991).
[25]K. Voyenli and E. Eriksen, "On the motion of an ice hockey puck," Am. J. Phys. **53** (12), 1149–1153 (1985).