Kharagpur Robosoccer Students' Group

Project Proposal

”Developing Collaborative Control of Multiple Robotic Agents in Adverse Environments through a Soccer Game”

Content

Introduction:

Problem solving in complex domains often involves multiple agents, dynamic environments, and the need for learning from feedback and previous experience. Robotic soccer is an example of such complex tasks for which multiple agents need to collaborate in an adversarial environment to achieve specific objectives. Robotic soccer offers a challenging research domain to investigate a large spectrum of issues of relevance to the development of complete autonomous agents. A soccer robot is a specialized autonomous mobile robot that is used to play variants of soccer. The robots should exhibit team strategies, dynamic decision making abilities and quick responsive behaviour. These traits can only be achieved by a harmony between mechanical, embedded and software aspects of the robot. Many leading universities and institutes have been working on implementing new and state of the art technologies in hardware and algorithms for path planning, team-work, intelligent decision making and innovative strategies. More and more work has been done in recent years to incorporate various forms of Machine Learning techniques to improve the accuracy and robustness of these robots. In general, building a successful team requires clever design, implementation and integration of many hardware and software sub-components into a robustly functioning whole making Small Size robot soccer a very interesting and challenging domain for research and education.

Aim:

The research group aims to work in two domains parallelly,

1. **Differential Drive Soccer Playing Robots -** These are a set of 6 differential drive design robots having a size limit of 7.5 cm * 7.5 cm * 7.5 cm. These bots are to be used to play a 3 vs 3 soccer match and test various path planning and hardware control systems. The challenges posed due to speed limits, varying physical variables and efficient play strategy when only three players are there in a team makes it an interesting research topic. There is scope for hardware improvements in terms of control and AI developments for better game play that we aim for.
2. **Omni Drive Soccer Playing Robots** - The robots having an omni wheel design has a restriction of 18 cm in diameter and 15 cm in height. The robot have dribbling and kicking skills - straight kick and chip kick making it complex in terms of bot-to-ball efficient energy transfer and path control. While dribbling a fast moving ball in the arena in one challenge, competing a team of 6 other bots at all three levels - skills, tactic and play is yet another challenge that has got the attention of researchers across the world. We are developing compact hardware for controls, efficient mechanical chassis and a world class game strategy.
3. **Humanoids -** Humanoids have been one of the most complex yet very interesting topic of research for everyone. Testing the humanoids across various adverse environments and platforms is still not completely explored. The humanoids developed in our project should be able to play a game of soccer. They should have efficient capabilities to walk (slow and fast), kick, stand up after falling and balance themselves in case of collisions.
4. **Simulation -** While working on the the above mentioned projects requires built-in state of art hardware, the initial testing and concept verification of all algorithms is done on simulators for

all the three robots mentioned above. With the simulators available(for all three) which are a reliable source for preparation and testing of the algorithms before deploying them on hardware. This makes our research efficient and cost effective.

Related Work:

Robosoccer has been a topic of interest to many researchers and robotics enthusiasts. It is here where different concepts from various verticals are used together. Various architectures have been proposed to prepare the hardware of the robot based on the small sized robots or humanoid. On artificial intelligence side, however, the path planning, game tactics remains somewhat the same since the key roles - goalie, defender and attacker remains the same. In the paper **FPGA Synthesizable Scaled ARM7 Soft Processor [1],** a subset of ARM 7, V4 instruction set is implemented to cater for FPGA based multiprocessor based SOC applications. A selected set of 32 bit instructions is implemented with single cycle datapath and random logic based instruction decoder. It removes obsolescence and limited scope owing to its complexity in larger modules. In other paper **Switching Techniques for Brushless DC Motors [2],** the researchers describe the different switching techniques to drive BLDC motors and presents the simulation of closed loop BLDC control. They have successfully implemented Trapezoidal Control on the BLDC motor using the gate driver circuit.The sinusoidal switching technique is not yet implemented. In our work, we have planned to observe the performance of both the switching techniques at required speed. The Vectorial switching technique will also be implemented as a part of our work.

On the mechanical side, **Optimizing a solenoid for a Robocup kicker [3]** paper describes the functioning of an electromagnetic actuator and methods to optimize it's working for the purpose of kicking ,as required. In this paper, based on the various control parameters such as the geometry and magnetic properties , they have used a simulation software FEMM to calculate and analyse the variables such as force, energy, heat production etc to finally reach an optimal design. They have reported the values of various parameters and constants for their design. **[4]** describes about DC/DC power converter that can raise the voltage of the output. In the scope of this research there is no need for power efficient topology since the objective is to store energy in capacitors and there is no equilibrium in the output to be met. The output will be fully capacitive and won't consume energy when it's charging. The PWM signal given to the MOSFETs should be optimized for reducing the charging time. The converter was not efficient, most of the time the converter will operate in Discontinuous Conduction Mode if the duty cycle is 50%. To tackle this situation a different control was applied to the basic boost converter. This change results in the current-mode boost converter. This converter consists in the same boost converter but does not have a fixed duty cycle, and instead, the fixed value is the maximum inductor current. Thus, they reported a faster charging time, with a charging rate much more constant (the boost operates always in CCM).

In the software level, Robot Operating System **[5]** framework is used. It provides operating system-like functionality like hardware abstraction, low-level device control, message-passing between processes and package management. STP(Skills, Tactics and Plays) Architecture is a hierarchical software architecture developed by CMDragons to control the autonomous team of robots in an adversarial environment is used as our software architecture. **[6]** SSL Vision is a shared vision system used as a primary sensor during the matches. It has been developed by a collaboration of large number of SSL teams. It is open-sourced for further development. One of the major tasks that need to be solved in robot soccer is path-planning. In the environment of this league, there are special requirements to a path-planning algorithm such as the strict time constraints and the highly dynamic obstacles. RRT(Rapidly - exploring Random Trees) is one of the most researched algorithms related to motion planning and undergoing improvements regularly. **[7]** Many teams now use upgraded versions with better heuristics and several other modifications of this algorithm. Tigers Mannheim have developed a RRT algorithm using waypoint cache and kd-trees in combination with DSS(Dynamic Safety Search) with outstanding results. A recursive algorithm was developed which outperformed RRT implementations of CMDragons and STOXs in terms of speed **[8]**.

In an attempt to understand humanoid design, the paper **Kinematics and dynamics modelling of the biped robot [9]** describes the kinematic and dynamic modelling of a biped. The biped trajectory and stability is characterized by equivalent force-moment and zero moment point. The paper illustrates the application of the techniques to plan the forward-walking trajectory of the biped robot. They have calculated the physical properties(like mass, center of gravity, moments of inertia) of a biped robot, Archie developed by IHRT. Also, using the Denavit-Hartenberg method and Newton-Euler equations, joint torques were obtained in terms of joint trajectories and the inverse dynamics were developed for both the single-support and double-support cases. They demonstrated how joint torques simulation can be implemented to plan the forward walking trajectory of a biped. They have devised an architecture that can be followed to simulate models, to plan the forward walking trajectory of a biped robot. This methodology is also helpful towards the selection of motors and link parameters for a biped.

One of the paper **[10]** described SCRAM role assignment method for the Humanoid gameplay. The state of art role assignment using SCRAM just got better with the use of prioritised role mapping and marking. In this paper the team describes its method to mark opponent bots and assign roles in a more intelligent way rather than just solving an assignment problem. Apart from strategy, faster and stable walking is of great importance when it comes to humanoids. Current research in Humanoids made it possible for humanoids to request continuous velocities in the forward, side and turn directions, permitting it to approach continually changing destinations (often the ball) more smoothly and quickly than unidirectional walking. The walk first selects a trajectory for the torso to follow, and then determines where the feet should be with respect to the torso location. The walk engine uses a simple set of sinusoidal functions to create the motions of the limbs with limited feedback control. The walk engine processes desired walk velocities chosen by the behavior, chooses destinations for the feet and torso, and then uses inverse kinematics to determine the joint positions required. More detailed mathematical approach is given in the paper **[11].**

Research Motivation

Small Sized Soccer robots:

Research on small sized soccer bots has been going on for the past 6 years. Different research groups have come up with different hardware & mechanical designs. However, several challenges in this field still exists. Adding to this fact, there are several verticals in which we can apply the basics of embedded electronics, mechatronics and artificial intelligence in the domain of robo-soccer to fill the research gap that still exists. The research is aimed to be extended to various industrial applications after it has been tested on a complex game like soccer. Swarm robotics and industrial automation are two such domains where our research have direct implications.

The novelty of our research lies in the approach we are taking to develop a team of 6 soccer playing robots from hardware end to software end. The circuit of the bots will be based on gate driver controlled BLDC motors with back emf as well as encoder control. The research intends to use various control loop algorithms to develop a robust hardware. In a similar vein, one of the other research challenges is to develop a design for chipping the ball in a parabolic path at maximum efficiency. This is one of the most challenging mechanical task that is being developed, as current chip kicker design, also known as push type chip kicker, transfers energy based on collisions in two stages, first between plunger and the chip plate and second between chip plate and the ball and thus incur significant energy losses. Also getting sufficient power from a flat type solenoid used for chipping, involves rigorous optimization of solenoid's parameters. While other state of art technologies report an efficiency of about 60%, we aim to develop a better design for it, by implementing a more optimized solenoid and a single collision based design for the chip kicker i.e. a pull type chip kicker, that doesn't rely on collision of plunger with the chip plate to transfer energy, thus reducing a substantial amount of energy loss.

Given the nature of this dynamic environment, the decisions need to be made in real-time. The bots are controlled via an overhead camera. It takes sixty frames per second which demands efficient computation. Delay is reported in the real time communication. While the gameplay of soccer is complicated, predicting opponent strategies is a challenging and a much needed task. Inter-tactic communication between the robots along with choosing an appropriate path planner based on the game scenario is extremely important for winning. For different game scenarios, we require different path planners, some being fast in computation and while others with more efficient paths.

Humanoids:

Increased demand for use of various types of robot have been of prime interest to us. Among the wide range of variety of robots, humanoid robots are the focus of researches nowadays. While a humanoid have been developed and is already being used in several domains, it's application and development in India is still limited. One of the very important reasons is it being expensive and the complex environment it will be dealing with while using it here. Our research aimed at developing a fully autonomous humanoid able to play a complicated game like soccer on our indigenous design is a step taken to overcome the aforementioned problems and contribute to the scientific community with a more robust design, a humanoid which can walk, run, kick and goal on its own. Our long term aim is to built a team of such robots that can challenge the real footballers.

Our research aims to build a fully autonomous humanoid robot which can play soccer and has the ability to reproduce human like movements, skills and strategies. Given the degrees of freedom associated with the bot(23), it becomes extremely difficult to make the bot execute the basics like standing and walking. It becomes extremely difficult to train the humanoid for walking/kicking given the number of parameter on which each skill depends. Due to restricted vision cone of the bots(120 degrees) there is some delay in tracking the ball position. This matters a lot when the opponent team has better walking speeds. Various training methods are available to improve bot speed out of which learning using Evolutionary Strategy makes the process fast and reliable. Having a fast computing algorithm for positioning is really necessary in order for the bot to compute other agent positions within the simulation cycle and send the message to the server. Hungarian algorithm is preferred for approaching this problem which has a complexity O(n^3). Apart from these, design analogous to the human body results in very high complexity and weight. Moving this weight requires even more powerful and heavier motors making the designing process a multi objective iterative optimization problem.

Challenges and Research Objectives

a. Development of Chip and Straight Kicker

Challenges Faced so far:

Till now the first stage of kicker circuit have been fabricated and tested. The difficulties faced by us are as follows:

i. Variable discharging using relays is not been fully controllable due to large switching time of Relays while IGBT's stop working after 5-6 cycles.
ii. The other major challenge is refilling the charge after every timestamp due to continuous charge loss from capacitors to environment when sitting idle. Thus it requires large power consumption.
iii. Also the other difficulty faced is the interference of charging circuit with radio frequency modules - transmitter/receiver on the other of the board.

The current version of solenoids used for chip and straight kicking faced major design challenges as follows:

I. The space available to implement a solenoid is limited in the bot and there are multiple parameters which defines a solenoid (like inner and outer diameter, length, thickness and material of solenoid, also number of turns and the wire diameter), so, optimizing the solenoid for maximum efficiency was major challenge.

II. The push type chip kicker's solenoid is a flat type solenoid which is more compact and thus provides lesser force than a cylindrical solenoid (implemented for straight kicker), thus we were not able to get satisfactory results from the chip kicker.

III. It was difficult to get the plunger hit the chip plate at its center of percussion which resulted in hinge reaction losses.

Future Research Objectives:

I. In future we aim to use a more robust controller for controlling switching duty cycle (variable switching) of charging circuit such as LT3750 for which we'll require DSO for obtaining waveforms.

II. We aim to incorporate more steps variable discharging through relays using interrupt based mechanism rather polling.

III. Break beam sensors will also be incorporated to ensure presence of ball to the bot , this will act as a double check mechanism and make our kicking system more robust.

IV. We will implement a more efficient pull type chip kicker using a cylindrical solenoid instead of the push type chip kicker with flat solenoid (which relied on impulsive collisions and caused significant losses).

b. Development of FPGA based control system

Challenges Faced with different architectures:

I. In architecture 1 when a single microcontroller is used for both speed calculation and control loop implementation an external interrupt is triggered in the system for each incoming tick from encoder. This load the system heavily and degrades the performance destabilizing the control loop at higher speed.

II. In architectures 2, when a separate microcontroller is used for speed calculation, the load on the central microcontroller is divided thereby making it possible for the system to control the motor for higher speed. The slave microcontrollers communicate with the master (central) microcontroller using UART communication protocol. The overhead caused due to this communication comes into action when target speed is above around 75 % of the maximum speed and the control loops become unstable. (Implemented in FIRA 2015)

III. While in architecture 3, when a FPGA is used for the same process, parallel processing is exploited and the system becomes capable of reaching maximum possible speed for the motors (capable of handling the ticks at maximum 6MHz). (To be implemented in SSL 2018)

The performance analysis of the different architectures is given below:

Architecture	Maximum achievable Speed	Maximum Ticks Frequency
Architecture 1	50% of maximum (5000 rpm)	32 KHz
Architecture 2	75% of maximum (7500 rpm)	48 KHz

Architecture 3	100% of maximum (10000 rpm)	6 Mhz

Future Research Objectives:

I. Architecture 1 and 2 are completely tested and implemented by fabricating custom circuits, while circuits are to be fabricated for the architecture 3.

II. In the third architecture, separate SoC are used for microcontrollers and FPGA. Few SoCs are available in market which house the power of both microcontroller and FPGA with dedicated area. Development of the system using these SoCs will improve performance.

c. Mosfets and Gate Driver based BLDC Control

Challenges faced so far

I. Initially we used a gate driver with a peak current of 1.5 A and maximum voltage rating of 18 V. Keeping delay time short and equal rise and fall time, we used Fairchild's Dual N-Mos P-Mos IC with high voltage and current ratings up to 40 V and 6.2 A current. The combination did not worked out well. The Dual N-Mos P-Mos IC burned out even when used with 18 V supply. We used current limiting resistors to prevent the current in the gate terminal of the MOS but the problem was not resolved. We believe that the surge current may be an issue.

II. Finally, we switched to discrete power mosfets by International Rectifiers. We designed the circuit again using the discrete mosfets and finally we were able to drive the motor using the ATMEGA2560 Microcontroller which was used to generate the commutation using the hall sensor feedback. The gate driver used initially were rated at exactly 18V and they burned out if we used a lower PWM value to drive the motor for longer amount of time.

III. Ultimately we decided to change the gate driver IC. Since we were using two discrete P and N channel mosfet, we faced the problem of *Deadtime* insertion. The rise and fall time of both the transistors are different resulting in a situation when both the mosfets are in ON state which results in a huge Spike of current from VCC to GND. This time was about 100 ns approximately. We looked for certain deadtime insertion method and finally ended up in using the RC Delay and AND -Gate propagation delay. We observed the possible gate driver inputs and made a logic level interpretation using AND gate and thus used its propagation delay. Then we tried to implement PID control on the motor but the RC Delay is creating some problems and the ticks obtained didn't showed any significant change when the RPM was changed.

   We decided to change the method of deadtime insertion and thought to inject deadtime using the microcontroller. We are successful in implementing closed loop PID Control with ATMEGA2560 but the frequency of ATmega2560 is not enough to inject the DEADTIME. Presently, we are trying to use the STM32f4 microcontroller to implement the closed loop control with proper deadtime.

Future Research Objectives

I. We have planned to test the circuits on ARM processors with proper PID control and deadtime insertion and then design the circuit on PCB and continue the integration with other parts of the bots.
II. Moreover, we want to use different control methods namely the sinusoidal and the back EMF control methods. We need to implement the sensorless back EMF method of BLDC motor control for the Dribbler Motor as they do not have any encoders.
III. We will implement more robust control loop on the motor using the Fuzzy logics and other control Algorithms.

d. Path planning and optimization

Challenges faced so far:

The robots in various sections of the field have very different roles to play and may require different types of planners. While a robot which defends the ball if front of the goal may require to move in a straight path most of the times, an attacker which needs to retrieve the ball from its own half may need to follow a clothoid path to extend its current inertia into a path that leads to an optimal retreating action. This has lead to the requirement of extensive study of planning algorithms for particular roles of an agent.

Future research objectives:

We propose to implement various state of the art path planning techniques that have been described in the above section with required modifications for better performance in robot soccer environments. We plan to implement the various path planning algorithms like RRT, VPF, and improve upon their results in highly dynamic environments.

I. Rapidly exploring Random Trees (RRT)**[17]** as described in **[Appendix-Section]** . Current use of RRT algorithm is computationally heavy due to requirements of continuous pruning and replanning in highly dynamic environments. RRT* , RRT connect**[18]** are some current algorithms that extend RRT in dynamic domains.
II. We propose to extend RRT for dynamic environments using prediction of obstacle motion to avoid repeated replanning. The repetitive behaviour of an agent in particular roles can be used to predict its motion by training a model to extrapolate a motion sequence. This can lead to the formation of time optimal paths which will need much less replanning. We plan to try various methods like Gaussian Process Regression and Spline Fitting to extrapolate the obstacle trajectories. We will also like to try a RNN based prediction using LSTM and Gated Recurrent units in our architecture for the same cause as this can be treated as a task of sequence prediction**[20] [Generating Sequences With Recurrent Neural Networks, Alex Graves].**
III. Apart from RRTs the above obstacle motion prediction will be implemented with other path planners like Virtual Potential Field **[20,21]**, Recursive planner**[19].** We would compare and benchmark all the path planners, in terms of planning time and choose the best planning algorithms suitable for particular roles of the agents.

f. Opponent Strategy Prediction:

Challenges faced so far:

For better formulation of strategies during a match it is of great help to be able to predict the actions which an opponent agent will be undertaking. This is a field less ventured yet in robot soccer.

Future Research Objectives:

This can be treated as an activity forecasting task in which we try to predict the action to be carried out by opponent agents in context.

I. This can be done through learning the motion pattern followed by clustering trajectories. Vanilla LSTM is agnostic to the behaviour of other sequences. Individual agents adjust their behaviour by implicitly reasoning about the motion of the neighbouring agents and opponent agents.
II. We expect the hidden states of LSTM to capture these time varying motion properties. In order to jointly reason about multiple agents we share the states between neighbouring LSTMs.
III. The hidden state h of the LSTM at time t captures the latent representation of the i'th person in the scene at that instant. We can share this representation with neighbors by building a hidden-state tensor.
IV. Given a hidden-state dimension, and neighborhood size, we can construct a tensor for the i th trajectory.
V. We can embed the pooled hidden-state tensor into a vector $\mathbf{a}^{t}_{i}$ and the co-ordinates into $\mathbf{e}^{t}_{i}$.These embeddings are concatenated and used as the input to the LSTM cell of the corresponding trajectory at time t **[Social LSTM: Human Trajectory Prediction in Crowded Spaces(Link)].** These motions can then be classified into particular behavioral categories which we plan to use for counter reactive strategy formulation.

g. Reinforcement Learning

Challenges faced so far:

In multi agent system where the agents have high degree of cooperation and coordination the decision making process if done in a deterministic way can be highly complex and impossible at times. But in case of systems like ours where the agents are fault tolerant to a certain extent non-deterministic way of decision making can suffice.

Future Research Objectives:

I. We plan to implement such non-deterministic decision making in form of reinforcement learning, Q-learning and other machine learning models. These will speed up the processes such as deciding each robot's next move taking into consideration it's fast paced changing environment and it's past performances.
II. These techniques have been used recently in game scenarios such as AlphaGo[reference]. What makes our case more challenging is that the reward to the Q function is delayed, i.e. until a goal has been scored or fended against. One solution can be using learning models with longer term memory, which is an open research problem. But learning with longer term memory can be challenging at times. The approach we plan on implementing is the variation with stack memory wherein the previous data of the states and actions are saved which can be used by the model.

III. Apart from the low-level decision making, such decision making methods can also be used as a high level strategy selector for the overall game. These two decision models will be highly correlated and interdependent, hence we also plan of finding a way to model this too into our learning techniques.

h. Robot Tracking:

Challenges faced so far

Tracking of robots/agents motion in a highly dynamic environment as in a soccer gameplay has a high degree of randomness. Extensive works have been done on efficient filters for such time-varying random process resulting in extended Kalman filters and particle filter.

Future Research Objectives:

We plan on adapting and building further upon these existing methods to adapt them to the highly realtime.

i. Humanoid Design

Challenges Faced So Far

I. While finding the D-H parameters, matlab code stopped due to lack of computation powers and redundant equations. These equations were computed again and again in order to correct the parameters but nothing was achieved till we corrected the transformation matrix.

II. Static Simulation results were derived first and they showed that the robot can stand only if we increase the foot area by 90%, but these would make it heavy and slow. Optimization of body structure was done in order to accommodate the humanoid in specified dimensions.

III. Static simulations results showed humanoid was not capable of walking unless hand was moved to an exact degrees. Finding these exact figures along with an equation was challenging and involved a lot time.

While finding the motor torques, 2 of the motors were in a state of breakage as the force and torque on them was much more than the tolerated limit.

Future Research Objectives

I. Development of humanoid will be achieved in two stages :

  A. Stage 1: Developing the lower body (Biped) - This is the most challenging part of humanoid development. In order to see what the simulation results lead to in real life, we need to implement first only the walking legs i.e. a biped robot in the hardware and see the exact movements and weight balance of them.

  B. Stage 2: Developing the upper body - After we are confident with the lower body, upper body of the humanoid robot will be made and the entire robot will be put to rigorous testing.

II. Static and dynamic analysis of each part individually: Analysis of various parts have been done to ensure they are within the tolerable limit. Designs need to be changed and analysis of each part has to be done carefully to find optimal relationship between them.

III. Reduction of material and looking for alternatives: Weight has to be reduced in order to make them more stable and easy to implement.

IV. Modification in design according to real life results: Analysis of real life results and then implement those in simulations and then again modify the prototype.

j. Humanoid Controls and Simulation

Challenges Faced so far

The CMA-ES algorithm takes 1-2 weeks to train and due to this we can’t scale it to every problem given the current resources we are using. With the advent of Deep Reinforcement Learning, the complexity would only increase.

Future Research Objectives

Currently we have used CMA-Evolutionary Strategy for improving low level skills, we aim to use deep reinforcement learning along with exciting new research in asynchronous actor-critic algorithms. :

I. Deep RL algorithms based on experience replay have achieved unprecedented success in challenging domains such as Atari 2600. However, experience replay has several drawbacks: it uses more memory and computation per real interaction; and it requires off-policy learning algorithms that can update from data generated by an older policy.

II. Instead of experience replay, one of the key insights is that you can achieve many of the same objectives of experience replay by playing many instances of the game in parallel.

III. Advantages of A3C: As our research problem motivates the parallelizing thing, we must start in this direction only, possible alternative are using evolutionary algorithms but CMA-ES is the state of art nonlinear evolutionary algorithm based optimizer and a huge amount of research has already been done with it. Another alternative being improving on simple RL based algorithms but with the OpenAI providing a simple simulation based environment this has already been looked upon.

Acknowledgement

We are sincerely thankful to the professors in- charge of the project.

- Prof. Jayanta Mukhopadhyay
  Department of Computer Science and Engineering
- Prof. Sudeshna Sarkar
  Department of Computer Science and Engineering
- Prof. Alok Kanti Deb
  Department of Electrical Engineering
- Prof. Dilip Kumar Pratihar
  Department of Mechanical Engineering

Their continuous guidance has been the only reason the group has been performing really well in past years. We look forward to their support for our research activities in future.

References

1. Design of 32-Bit RISC Processor and efficient verification. Geun-young Jeong, Ju-sung Park Dept. of Electronics Engineering, Pusan National University San-30 Jangjeon-2 dong, Geumjeong-gu, Pusan 609-735, Korea
2. S. A. R. Sierra, J. F. M. Carballido and J. L. V. González, "Switching techniques for brushless DC motors," CONIELECOMP 2013, 23rd International Conference on Electronics, Communications and Computing, Cholula, 2013, pp. 162-166.doi: 10.1109/CONIELECOMP.2013.6525779
3. Design and Implementation Solenoid Based Kicking Mechanism for Soccer Robot Applied in Robocup-MSL in International Journal of Advanced Robotic Systems,Vol. 7, No. 4 (2010) 69 ISSN 1729‐8806, pp.73‐80
4. ROS: an open-source Robot Operating System, Morgan Quigley, Brian Gerkey, Ken Conley, Josh Faust, Tully Foote, Jeremy Leibs, Eric Berger, Rob Wheeler, Andrew Y. Ng. To Appear in Open-source software workshop of the International Conference on Robotics and Automation (ICRA), 2009.
5. Ball Handling Mechanisms for Mobile Robots Thesis by Miguel Santos Serafim, October 2013
6. STP: Skills, tactics and plays for multi-robot control in Adversarial Environments by Brett Browning , James Bruce , Michael Bowling , Manuela Veloso in IEEE Journal of Control and Systems Engineering, 2004
7. Resolution complete rapidly-exploring random trees. P. Cheng and S. M. LaValle. In Proceedings IEEE International Conference on Robotics and Automation, pages 267--272, 2002.
8. Fast Path Planning Algorithm for the Robocup Small Sized League by Saith Rodriguez, Eyberth Rojas, Katherin Perez, Jorge Lopez, Carlos Quintero, and Juan Calderon in RoboCup Symposium, 2014
9. X. Bajrami, A. Dermaku, A. Shala, and R. Likaj, "Kinematics and dynamics modelling of the biped robot," in IFAC Proceedings Volumes (IFAC-PapersOnline), vol. 15, no. PART 1, 2013, pp. 69-73.
10. SCRAM: Scalable Collision-avoiding Role Assignment with Minimal-makespan for Formational Positioning. Proceedings of the 13th International Conference on Autonomous Agents and Multiagent Systems (AAMAS), Paris, France, May 2014.
11. Design and Optimization of an Omnidirectional Humanoid Walk: A Winning Approach at the RoboCup 2011 3D Simulation Competition. Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI-12), Toronto, Ontario, Canada, July 2012
12. KIKS 2015 Team Description Paper - Juan Pablo Mendoza, Joydeep Biswas, Danny Zhu, Richard Wang, Philip Cooksey, Steven Klee and Manuela Veloso in Robocup 2015
13. Anorak Team Description Paper titled - “Advancements and Attack Strategy Comparison in the CMDragons 2014 SSL Team” - Hamza Munir and Aamir, Robocup 2015
14. CMDragons 2015: Coordinated Offense and Defense of the SSL Champions, Juan Pablo Mendoza, Joydeep Biswas, Danny Zhu, Richard Wang, Philip Cooksey, Steven Klee, and Manuela Veloso In RoboCup 2015: Robot World Cup XVIX, Springer, January 2016.
15. Detecting and Correcting Model Anomalies in Subspaces of Robot Planning Domains, Juan Pablo Mendoza, Manuela Veloso, and Reid Simmons. In Proceedings of AAMAS'15, the Fourteenth International Joint Conference on Autonomous Agents and Multi-Agent Systems, Istanbul, Turkey, May 2015.

16. Opponent-Driven Planning and Execution for Pass, Attack, and Defense in a Multi-Robot Soccer Team, Joydeep Biswas, Juan Pablo Mendoza, Danny Zhu, Benjamin Choi, Steven Klee, and Manuela Veloso. In Proceedings of AAMAS'14, the Thirteenth International Joint Conference on Autonomous Agents and Multi-Agent Systems, Paris, France, May 2014.
17. Sampling-based Algorithms for Optimal Motion Planning by Karaman and Frazzoli in International Journal of Robotics Research, Pages 846-894, Volume 30 Issue 7, June 2011
18. Kuffner J., LaValle M. "RRT-Connect: An Efficient Approach to Single-Query Path Planning". Proceeding of the 2000 IEEE International Conference on Robotics and Automation, 2000.
19. Fast Path Planning Algorithm for Robocup SSL League by Rodriguez et al. Springer International Publishing Switzerland 2015 R.A.C. Bianchi et al. (Eds.): RoboCup 2014, LNAI 8992, pp. 407–418, 2015
20. Zhu. Q, Yan Y, and Xing Z. "Robot Path Planning Based on Artificial Potential Field Approach with Simulated Annealing". Sixth International Conference on Intelligent Systems Design and Applications, Jinan, China, 2006,2,pp.622-627.
21. C. Qixin, H. Yanwen, and Z. Jingliang, "An evolutionary artificial potential field algorithm for dynamic path planning of mobile robot," in Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on, Oct. 2006, pp. 3331-3336
22. Tomás Lozano-Pérez , Michael A. Wesley, An algorithm for planning collision-free paths among polyhedral obstacles, Communications of the ACM, v.22 n.10, p.560-570, Oct. 1979

Appendix I : Proposed Budget

Purpose	Particulars	Quantity	Price per unit	price
FPGA Development	Spartan 6 (FPGA Ics)	15	3,456	51,840
	Circuit fabrication	15	2,000	30,000
	Circuit Components	15	2,000	30,000
	Zynq development Board (zybo)	1	23,737	23,737
	Oscilloscope (50 MHz 2 Channel)	1	24,270	24,270
	Total			**1,59,847**
Kicker Development	2200uF Capacitors	30	1,250	37,500
	Circuit fabrication	15	2,000	30,000
	Circuit Components	15	3,000	45,000
	Break Beam sensors	15	1,000	15,000
	LT3750 controller	5	600	3,000
	Total			**1,30,500**
BLDC Control Development	Gate Driver	150	130	19,500
	Mosfet p channel	150	65	9,750
	Mosfet N channel	150	90	13,500
	STM32F407	60	1,100	66,000
	Circuit Components	20	3,000	60,000
	Circuit Fabrication	20	2,000	40,000
	Total			**2,08,750**
	Prototyping body	-	-	1,55,000

Mechanical Design	Prototyping (dribbler and kicker)	-	-	70,000
	Final Bot Manufacture	7	1,40,000	9,80,000
	Total			**12,05,000**
Humanoid Development	Motors	25	45,000	11,25,000
	3 D Printer	1	2,00,000	2,00,000
	Electronics	--	1,50,000	1,50,000
	Manufacturing	--	3,00,000	3,00,000
	1 Nao Bot to be purchased	1	10,00,000	10,00,000
	Nvidia TItan X(P) or AWS	1	1,00,000	1,00,000
	Misc + Camera	--	2,00,000	2,00,000
	Total			**30,75,000**
Regular Consumables	Batteries	40	2,500	1,00,000
	Motors	12	17000	2,04,000
	Misc	--	--	1,50,000
	Total			**4,54,000**
Travel and Participation (3 Year)	Contingency	3	50,000	1,50,000
	Travel	3	50,000	1,50,000
	Total			**3,00,000**

Proposed Budget = 55,33,097

Appendix II : Year Wise Deliverables

Current	**Small Sized Soccer Playing Robots** ● 6 bots are all-ready for a match based on ARM controllers ● Being used for testing path planners and controls of the bot ● The code is in complete running condition and integrated with SSL Vision ● MergeS curve is used as the path planner
	Humanoid ● The solidworks design for the humanoid simulation is ready. ● Static stress analysis is complete.
Year 1	**Small Sized Soccer Playing Robots** ● Built in-house BLDC controller using gate drivers and mosfets ● Optimize kicker charging time and incorporate variable kicking ● First prototype for FPGA based control circuit and redesign the kicker circuit ● Increase the speed of the bots and fabricate new circuits ● Completion of mechanical prototyping of chip kicker ● Change code architecture to accommodate multiple path planners ● Use an improved ROS library OMPL to bring this change ● Increase the speed of the bots and fabricate new circuits ● Referee Box ROS package to be completed. ● Integration of chip kicker to the plays for long range shots. ● Testing for errors and stability for Robocup SSL in the month of May and June.
	Humanoid ● Dynamic simulation, analysis and prototyping of biped, ie the lower body of humanoid. ● Rigorous testing and further design modifications.
Year 2	**Small Sized Soccer Playing Robots** ● Design a fabricated prototype FPGA circuit for full bot control ● Get a new set of 6 robots with mechanical optimization and chip kicker implemented ● Chip-distance will be calculated accurately using various algorithms. ● Reinforcement learning algorithms will be implemented for Play selection and Role assignment using python libraries. ● Further development of the game play
	Humanoid ● Development of upper body of humanoid (Design, simulation & Prototyping) ● Completely ready Biped and controlling it via the AI
Year 3	**Small Sized Soccer Playing Robots** ● Prepare 6 circuits based on FPGA

	● Extend that research to optimize the computation the spartan board. ● Migration to ROS 2.0 to make use of the latest features introduced in the most recent ROS framework. ● Improved Positioning play to be implemented for smoother, faster and more efficient movement of robots. ● Improved skills, tactics and plays to be implemented.
	Humanoid ● Full Humanoid Prototyping and testing ● Debugging errors and further developments

Appendix III : Year Wise Budget Requirement

Year	**Particulars**	**Budget**
Year 1	--------------------	**29,89,726**
Year 2	--------------------	**20,58,723**
Year 3	--------------------	**4,84,648**
Total	------------------------------	**55,33,097**
Year 1	FPGA	27,726
	Kicker & BLDC Controls	2,72,500
	Small Sized bot mech design	2,25,000
	Humanoid Mechanical Development	10,75,000
	Humanoid Control Simulation	11,00,000
	Travel and Contingency	1,00,000
	Regular Consumables	1,89,500
	Total	**29,89,726**
	FPGA	72,473
	Kicker & BLDC Controls	66,750

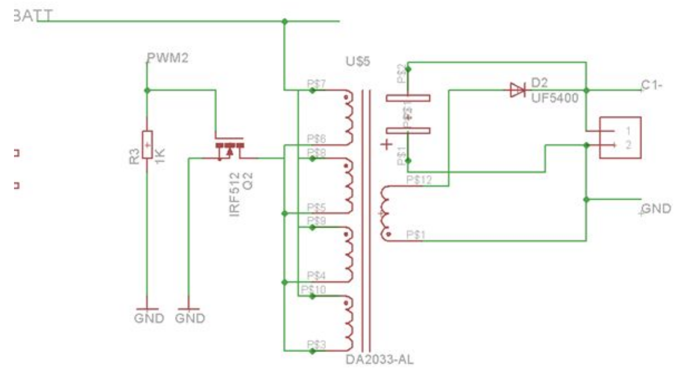
Year 2	Small Sized bot mech design	9,80,000
	Humanoid Mechanical Development	6,50,000
	Humanoid Control Simulation	0
	Travel and Contingency	1,00,000
	Regular Consumables	1,89,500
	Total	**20,58,723**
Year 3	FPGA	59,648
	Kicker & BLDC Controls	0
	Small Sized bot mech design	0
	Humanoid Mechanical Development	2,50,000
	Humanoid Control Simulation	0
	Travel and Contingency	1,00,000
	Regular Consumables	75,000
	Total	**4,84,648**

Appendix IV : Development of Chip and Straight Kicker

Straight and Chip Kicker Circuit Design Based on Flyback converters

The aim is to develop robust charging and discharging circuits to implement straight and chip kicker mechanism. The charging circuits based on Switch Mode Power Supply (SMPS) is shown below.

Charging circuit is based on flyback converters. DC-to-DC techniques use transformers that work at much higher frequencies. Power MOSFETs are used for switching which are controlled by a PWM signal from Atmega328 micro-controller with a frequency of 200 Khz. The PWM duty cycle is optimized for fast charging of capacitors. The capacitors can charge up to 150V in 6-7 seconds. The continuous feedback of charged capacitors is monitored by microcontroller using ADC peripheral. Two capacitors are connected in series with specifications of 2200uF each with a rating of 250V.

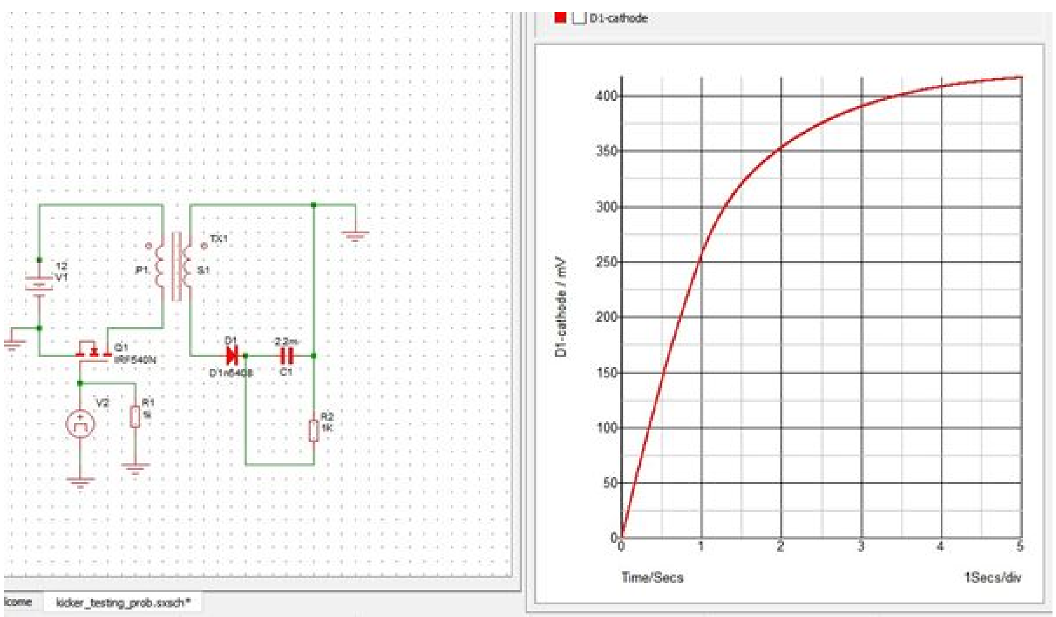


In the **Discharging circuit,** the solenoid is connected in parallel to capacitors in series with relays for switching the discharging circuit. The relays are also controlled with GPIO of microcontroller. Two relays have been used in parallel for handling the high discharging current.

Simulation of Charging Circuit in SIMPLIS

The charging circuit is simulated by the varying all possible parameters i.e no. of turns of transformer, duty cycle of switching and the plots are obtained. The best results obtained are taken into account while fabrication of circuit.

Topologies used for switching MOSFET in charging circuit:



Constant duty cycle :-Circuit takes comparatively large amount of time to reach steady state but less complex in implementation.

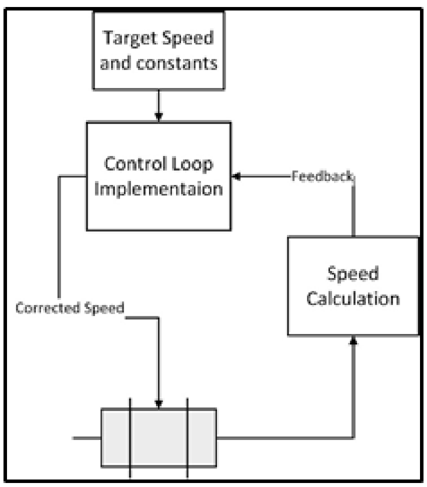
Discrete Steps duty cycle:- Circuits converges to steady state for only particular steps of duty cycle and that is quite random. Thus hit and trial is required.

Proportional control: The duty cycle is varied by taking feedback from capacitor voltage using a

Proportional control. Theoretically, its best way but charging time is almost same as in case of constant duty cycle.
Elements used for discharging circuit so far are :
IGBT: Integrated Bipolar Junction Transistor is chosen because of its capability of fast switching along with large collector current. But it requires robust protection circuits i.e Snubber. Due to improper snubber design, IGBTS stop working just after 5-6 cycles.
Relays: More robust than IGBT's and simple to control. But large switching time thus variable discharging is not exactly controlled.

Appendix V : Development of FPGA based control system

PID and Fuzzy Based Control loop Implementation using FPGA



To control the velocities of the BLDC motors fitted with encoders, a PID and Fuzzy based control loop is formed. The detailed flow diagram of the control loop implementation is given below.
Speed calculation involves the estimation of current speed and direction of the motor using the pulses coming from the high resolution encoder fitted with motor. The control loop implementation block monitors the speed of the motor and calculates corrected speed using target and precomputed constants. This corrected value is fed to the motor using a motor driver. With proper synchronization of all the events a real time monitoring and controlling of the motor speed is achieved.
There are multiple architectures we have used to implement the same, which are mentioned as follows:
Architecture 1: A single microcontroller common speed calculation and implementing control loop
Architecture 2: Multiple inferior microcontrollers for calculating the speed of motors and a central superior microcontroller
Architecture 3: A FPGA controller for calculating the speed and implementing PID using the constants precomputed from a central microcontroller39

Appendix VI : Mosfet and gate drive BLDC based control

The circuit comprises of three gate drivers driving three independent discrete P and N MOSFETs inverter circuit. Moreover the basic power circuit consists of fuse (3 Amp) and a main switch along with an LED. Trapezoidal control using hall sensors feedback is used to drive the BLDC motor. The commutation table is generated using a microcontroller feeding the PWM signal to the gate drivers input which in turns switches the MOSFETs depending on the hall sensors values.

Appendix VII : Path Planning and Optimization

Architecture:-

We use a 3 layered architecture to generate responsive team behavior which is adaptive to dynamic changes in the field during the execution of the soccer match. We have a centralised system, which tries to incorporate the benefits of a de-centralised system with the ease of a centralised system. For that we shifted our code-base from normal C++ architecture to ROS(Robot Operating System) based architecture.

ROS has been used in various other robotic arenas but is a new addition to the scene of robot soccer. Not many teams have tried to use this as a base for their architecture. Most of the teams use a C++ or Java based architecture on the lower level followed by a higher level architecture in Python to incorporate machine learning techniques.

Current Architecture:

Basler ACA2000 165uc, USB-3 cameras have been integrated with SSL-Vision which did not have the support for USB-3 cameras earlier. Using the drivers provided by Basler and some changes in the source code of SSL-Vision provided by Robocup, we have been able to integrate these cameras.

The Nodes :

There are separate packages in ROS, which execute on separate threads to generate a concurrent teamwork.

SSL-Vision publishes the vision information (Position of robots and ball) using Google Protobuf messages which is received by our the node **Vision_Comm**. This is a ROS node responsible for receiving the Vision Message packets and converting them into ROS messages which are published to **vision_data topic** for further processing. This node runs in two modes, one for testing in Simulator Mode which receives the data from GRSIM(a simulator provided by ROBOCUP for testing of the codes), and secondly in vision mode.

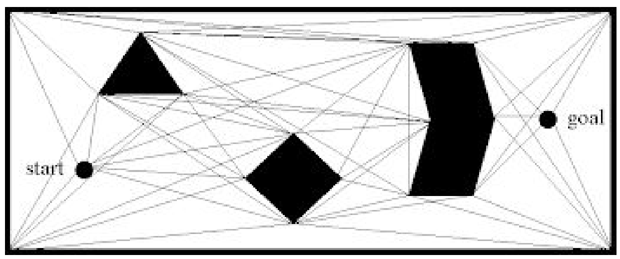
The **belief_state node** is another node which subscribes to the vision_data topic and is responsible for creation of a world model from the raw data. A low pass filter is first applied to raw data to smooth out the inconsistency in the in incoming data. After trying out Kalman Filter and Savitzky Golay, we found out the later one to be faster to converge, which is needed in a highly dynamic situation as is in our case. Then calculations like robot and ball velocities, various field predicates(ball_in_our_possession, free_ball, etc.) are made from the filtered data. This data is published on the **belief_state topic** for other nodes to use.

This belief_state topic is subscribed by the **6 robot nodes** and the **play_node.** The play node is responsible for the higher level strategy formation of the robots. The play node uses the play library to generate roles for each robot based on the state of the game. These roles are published on the the **robot_id topic**. Each robot node subscribes to its corresponding robot_id topic and executes the roles assigned to it using the tactic library which in turn uses the skill library. Each robot node publishes the velocities the calculated velocities (vx, vy and w) to **bot_comm topic**.

The **nodes bot_comm/grSim_comm** subscribe to the bot_comm topic and convert the vx,vy and w values into the velocities of the 4 omnidirectional wheels. These wheel velocity values are finally transmitted from the robot_comm node via the communication modules to the robots.

Path Planning :

Visibility Graph**[22]**:-



In terms of path generated this planner gives the most optimized path. It uses the corners of all the obstacles which includes HomeTeam Bots, Opponent Bots and all corners in the boundary and goal lines of the field. Using these points A* or Dijkstra's algorithm is used to find the shortest path. After the generation of the path curve fitting is done to generate a curve that uses the points to form a trajectory. It was tested and found that it was very slow and unstable in dynamic environment.

RRT Connect**[18]**:-

It is used by a majority of teams (including modifications) in Robocup SSL League. The algorithm is probabilistically complete. RRT algorithm grows a tree with its root at the starting position by sampling random points from the search space (i.e. arena). As each sample is drawn, a connection is

attempted between it and the nearest state(node) of the tree. If the connection is feasible then a new state is added at a certain fixed distance from the nearest state. Due to uniform sampling of the search space, the probability of expansion of the tree towards a large unsearched space is higher. The downside is that uniform sampling produces unnecessarily long paths.
Future Scope**:** Better heuristics and implementations(VF-RRT, RRT* etc.) of the above algorithm can be used to produce better paths.

Recursive Planner**[19]**:-

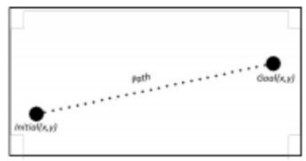
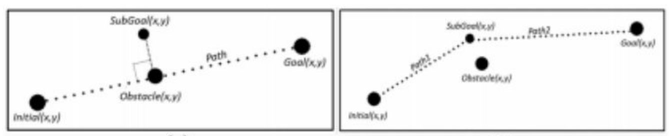


Fig. 3. Path without obstacles



(a) (b)

Fig. 4. (a) Subgoal Selection. (b) New paths created with subgoal

Since there was a requirement of a fast path planner that gives shorter paths than RRT, we tested Recursive Planner. It first tests if there is an obstacle in between the HomePos and destination. If no, then it directly joins the points else it samples a point that lies on the perpendicular bisector of the line joining HomePos and destination at a fixed distance k. Then the algorithm is recursively applied to the 2 subsections generated (i.e. - HomePos, generated point and generated point, destination). Upon testing it was found that algorithm was very fast and gave short paths.
The downside of this planner is that their was a huge probability of replanning and the path changed frequently. Though path generation was quick but the algorithm performed slow due to the problem of replanning at every step.
Future Scope: The algorithm can be modified to ensure that the frequency of replanning is restricted.

Integrating learning and planning: We integrate learning and motion planning for a soccer playing robot using Bayesian optimisation. Trajectories generated using end-slope cubic Bezier splines are first optimised globally through Bayesian optimisation for a set of candidate points with obstacles. Then, trajectories and the corresponding prior are stored in a database to be queried for real time path optimisation. The closest planning situation is identified using k-Nearest Neighbour approach with robot and obstacle position, and velocity as features. It is further optimised online through reuse of prior information, enabling optimised trajectories with obstacle avoidance in high-velocity dynamic soccer game. A velocity profile for the trajectory is generated incorporating kinodynamic constraints of the robots, using arc-length reparametrization through a secondary spline. Trajectories are tracked, which triggers trajectory replanning in case of large displacements. Extensive testing is done on developed simulator, as well as on our robots. Our method shows marked improvements in mitigating tracking error, and reducing traversal and computational time over competing techniques under the constraints of performing tasks in real time. This work in under review at Advances in Robotics 2017.

Appendix VIII : Mechanical Design

Dribbler

A dribbler bar made of Aluminium is used. Dribbler bar is directly driven by a brushless Maxon EC16 (16W) motor through two external spur gears with ratio of 4:1.

Analysis of dribbler design

The design has slots or grooves on the dribbler surface to prevent slippage of the ball during turning of the bot. The equation for dribbler height was determined using the MATLAB code for different percentage of area coverage. We found out the various heights of the dribbler. The MATLAB code was written and practically verified to optimize the height of the dribbler. We also had to consider the angle of the chip kicker in determining the height as the normal to plate of the kicker should be parallel to the common tangent between ball and the dribbler. Thus the optimum height of the dribbler = **3.97 cm** (from base to center of dribbler)

Percentage of area covered = **18 %**

Solenoid

The solenoid mainly consists of 3 parts: the plunger, the shield and the coil. The plunger is the movable part which delivers the force to the leg. The shield makes sure that the magnetic field doesn't influence other systems in the robot, like the motors or the laptop and also makes sure that the other systems don't influence the magnetic field built by the coil. The shield also decreases the reluctance at the outer positions of the solenoid. To be sure most energy will be available inside the coil, the reluctance outside the coil has to be as little as possible. The reluctance of a steel shield is much less than the reluctance of air. So the shield makes the solenoids also more efficient.

Straight Kicker

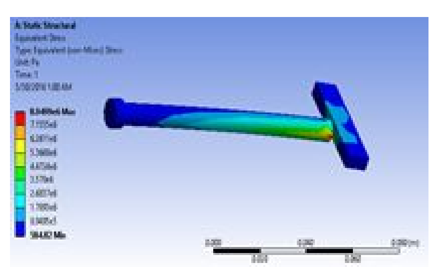
The straight kicker is powered by two 250V 2200uF capacitor, which can be charged by a step-up converter upto 200V each but we have limited the voltage to 150V to limit the kicking speed to 6 m/s to comply with SSL rules. After electromagnetic analysis using FEMM 4.2 and MATLAB, we found that 23 AWG wire gave the optimum results. So 23 AWG wire is wound with 870 turns.

Estimation of maximum Force on straight kicker

For the estimation of the maximum force on the kicker during the collision with the ball, ADAMS simulation was used. We got the maximum deceleration of the front part of the kicker and multiplying it with the mass of kicker we got the required force.

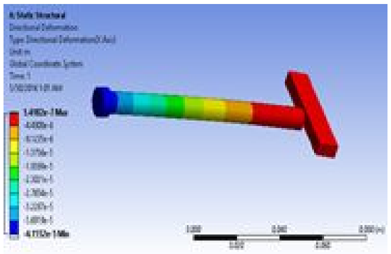
Maximum deceleration = 4.1×10^{3} ms^{-2} (approx)

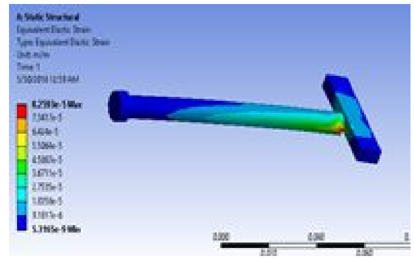
Mass of the kicker = 64.35 gm



Maximum force = 263.835 N

Total deformation

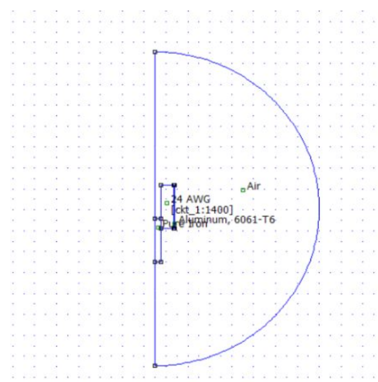




Analysis of chip kicker

The new chip kicker with pull type mechanism was designed and the previously used flat solenoid was replaced by cylindrical solenoid and pulling action was achieved by the plunger of the straight kicker.

FEMM MODEL OF SOLENOID

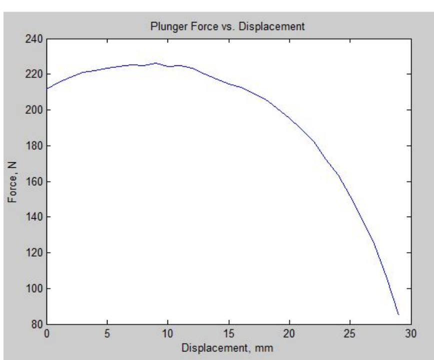


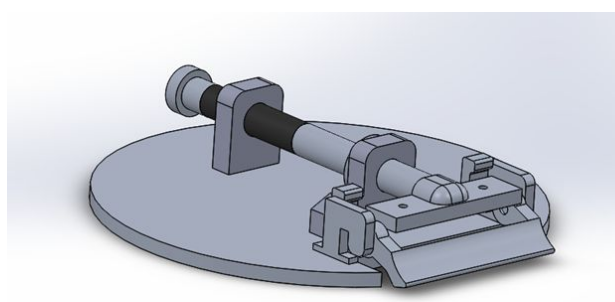
Due to increase in space in the new model we increased the diameter of the solenoid and consequently increased the no. of turns. We used **24 AWG** wire and wound the solenoid with **1400 turns**. Aside is the FEMM model of the solenoid. A constant 40 A current was taken for finding out energy in FEMM .

The results are as follows

Total energy transferred to the solenoid = **5.726 J**

The plot of plunger force vs displacement graph is as follows





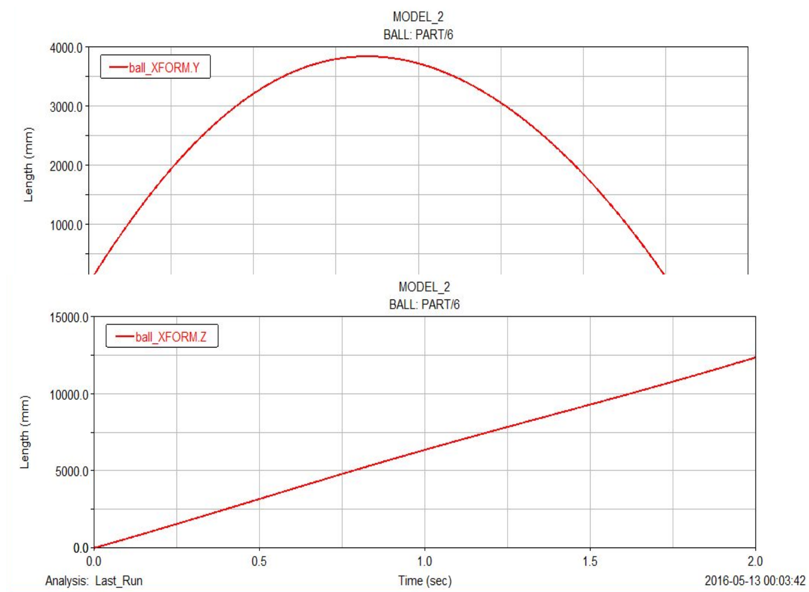
After obtaining the graph we got the expression of force with respect to displacement of plunger. Feeding this expression in ADAMS we obtained the range and max height of the ball.

We performed a series of tests by changing the wire diameter and following are the results in **ADAMS SIMULATION** -

The RANGE OF THE BALL **=** 10 m

The MAX HEIGHT OF THE BALL = 3.1 m

Then we changed the **24 AWG** wire with **22 AWG** wire. So the no. of turns reduced from **1400** to **930.** The plot obtained from MATLAB was a little better and then it was practically tested. Energy transferred to the plunger = **3.727J.**



Results of this design yielded :-

Range of the ball = 6.4 m (approx.)

Height of the ball = 2.7 m (approx.)

The **24 AWG** wire was then replaced by **26 AWG** wire. The no. of turns increased from **1300** to **2200 .**

Energy = 9.146 J

Range = 9.1 m.

Max height = 2.75 m.

We also performed tests with other wire diameters ranging from 18 AWG to 28 AWG. From the results we found out that **23 AWG** gave the best results which are as follows-

Energy Transfer = 5.92 J

Range of the Ball = 10.83 m

Height of the Ball = 3.58 m

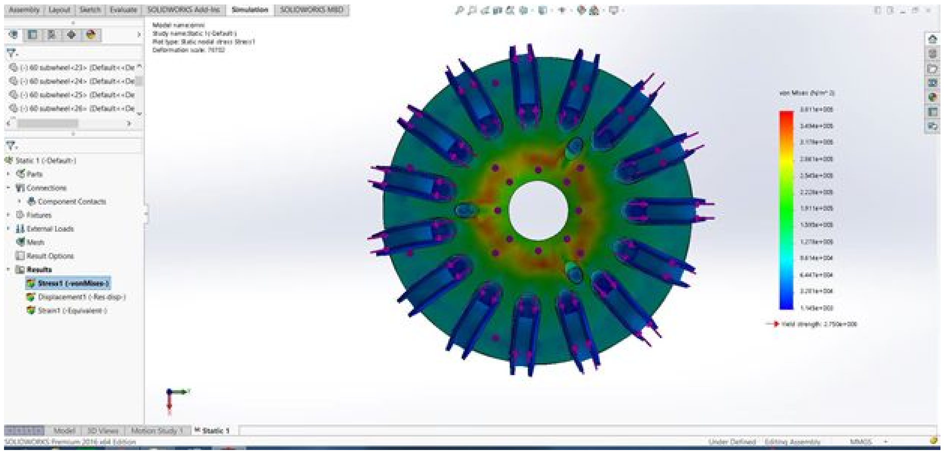
Thus 23 AWG wire was implemented in all the solenoids.

Wheels

The number of sub-wheels in the omni wheel was optimised by analyzing the stress sub-wheels might experience during a collision. Another major change in our older robot is that the screw joining the wheel and the shaft used to loosen after few minutes of continuous motion of wheel. So we redesigned the washer and implemented a new square locker mechanism to prevent the loosening of screw

Stress Analysis

Analysis of Omni Wheel in SolidWorks

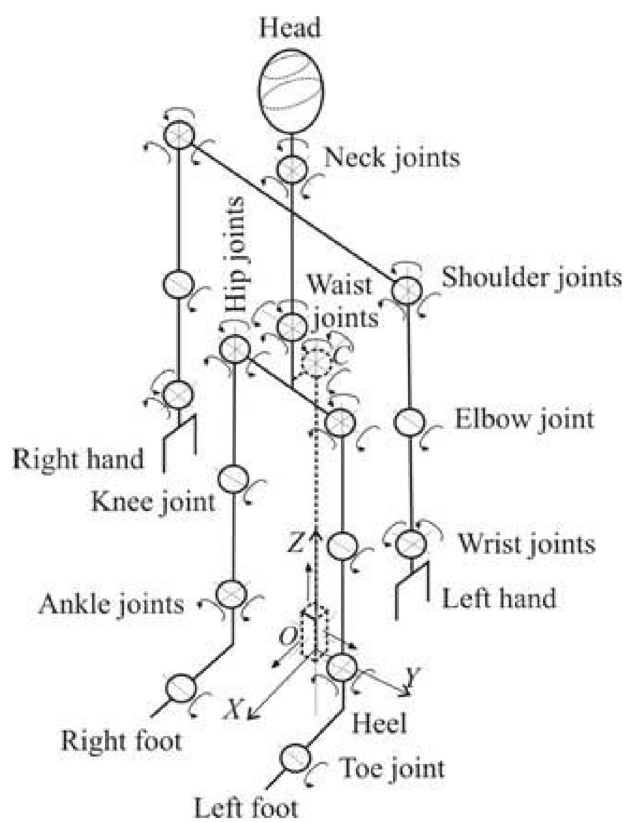


Stress Analysis of the body was performed prior to manufacturing and it was ascertained that the stresses were coming under tolerable limits , even when collisions occurred.

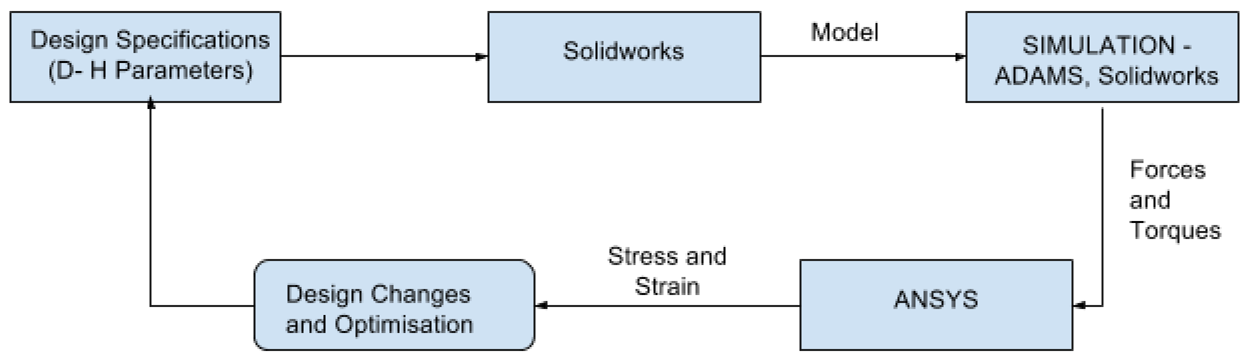
Appendix IX : Humanoid Design

Robot kinematics which basically deals with study of kinematics of open chain mechanisms. It also studies the relationship between the dimensions and connectivity of kinematic chains and the position, velocity and acceleration of each of the links in the robotic system, in order to plan and control movement and to compute actuator forces and torques. The relationship between mass and inertia properties, motion, and the associated forces and torques is studied as part of robot dynamics. To maintain dynamic balance during the walk, a robot needs information about contact force and its current and desired motion. The solution to this problem relies on a major concept, the Zero Moment Point (ZMP) and the Inverted Pendulum Model. Robot Kinematics can be broadly classified into Forward Kinematics and Inverse Kinematics.

The figure beside shows the kinematic diagram of a human giving an overview of the degrees of freedoms of various joints and their axis of rotation.

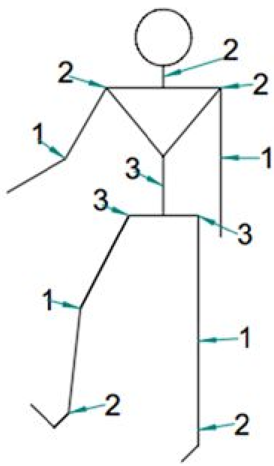


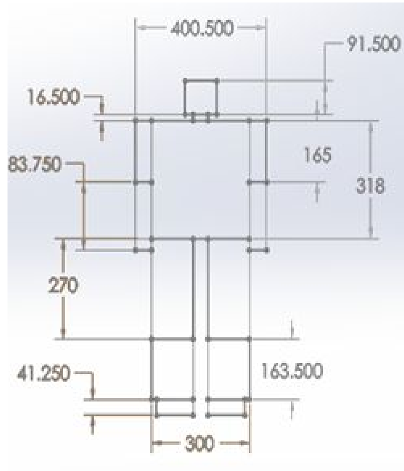
The workflow of designing implemented by us is represented below and one can notice that it is an iterative process, thus we can evolve our design to produce a more optimized and stable design.

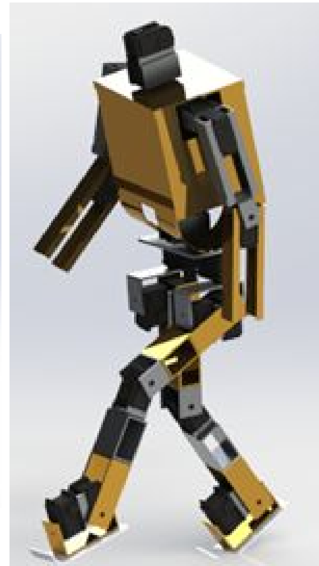


We designed a full humanoid robot and performed analysis of stresses on various joints, thus giving us the required data to decide upon the motors required. The designing process involves the use of multiple softwares (like SolidWorks, Adams, Ansys, Matlab) to get the desired model. The various steps are explained below-

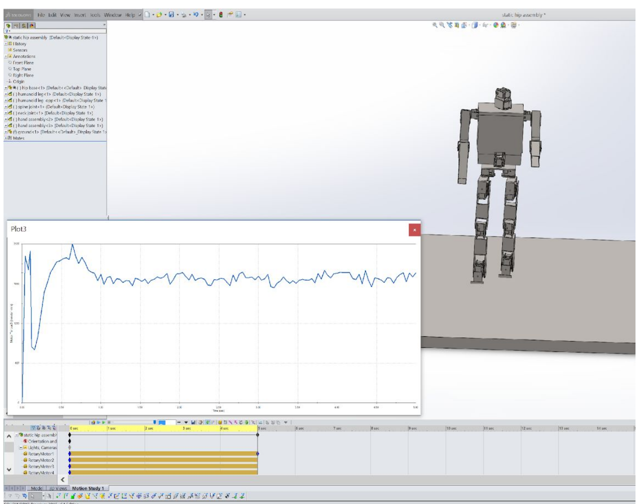
1. **D-H Parameters:** At first the total no of degrees of freedom is decided analogous to human body. The current design specifications are as follows-
    a. Degrees of freedom = 23
    b. Height = 90 cm
    c. Width = 40cm
    d. Weight = 7.5kg (including electronic components)
    Then the various D-H parameters are calculated which further gives us the transformation matrix of each of the individual links.
2. **Solidworks Model**: On deciding the basic parameters we designed the model in solidworks. Which would further be used for static and dynamic simulation.
3. **Simulation and Torque Analysis:** The complete analysis was done in solidworks and the optimization of various parameters was done. The optimized results are shown in the pictures. As foot is raised the torque increases but the bot is able to balance itself. The bot is completely stable and it proves the bot is capable of withholding standing stresses.

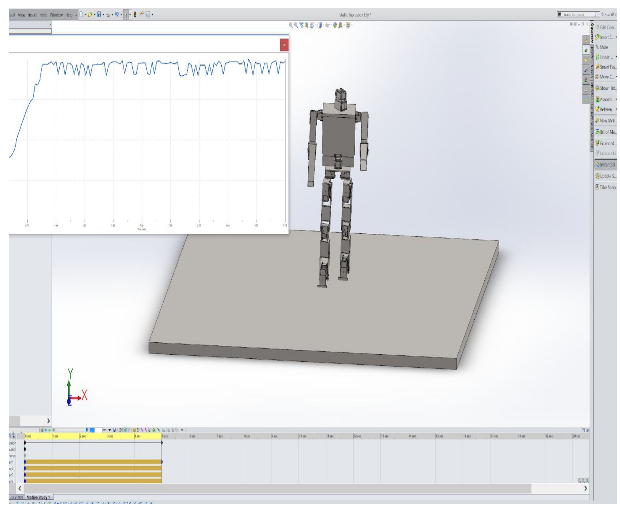






Case 1: Standing Torques

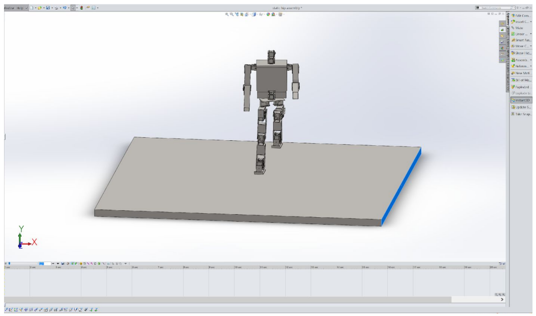


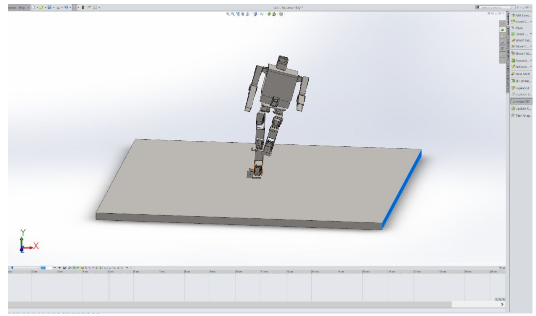


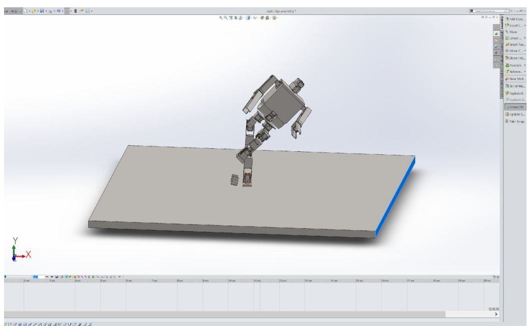
This is the calculated torque for a test Humanoid Robot to keep standing. (for the left leg of the robot). This is the calculated torque for the same position as the first case. (For the right leg).

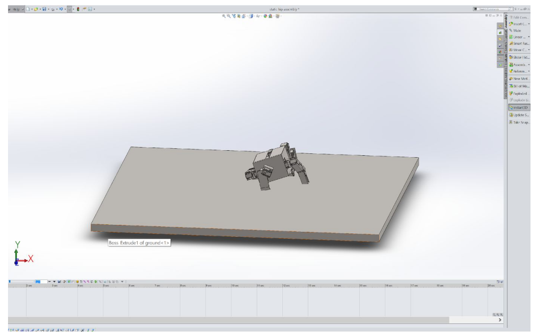
Just in case things are not clear the peak torque is 3.4 N.m and 2.5 N.m respectively. It is not same because we weren't able to make the robot to stand in exact upright manner by freeing all joints due to complications arising for other movements. We will try our best to give values for better positions (more upright).

Case 2: Falling torques









The above pictures represent the falling procedure. The fall was pretty bad . We calculated the torques for the ankle motor which maximum torques in the above fall.

Fig aside

The torque for the initially raised leg was low initially since it was under free fall. It suddenly peaks when it hits the ground whose value was 2.29 N.m.

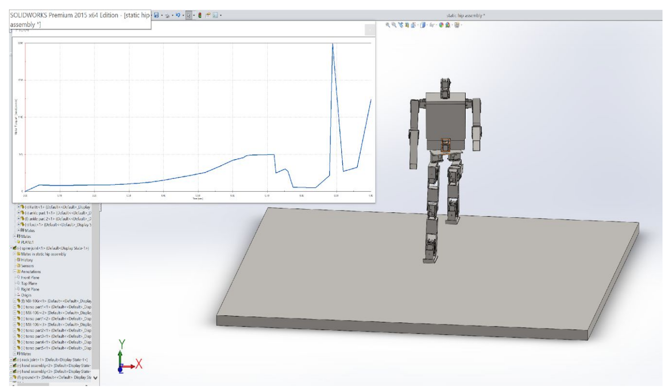
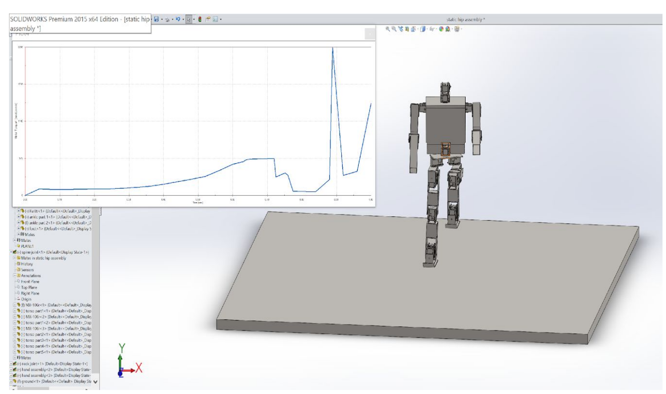


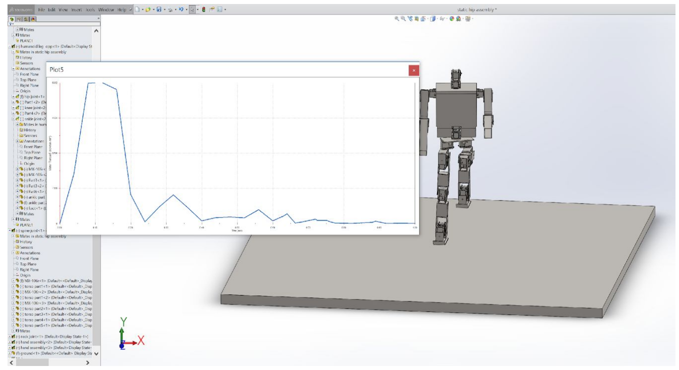
Fig aside

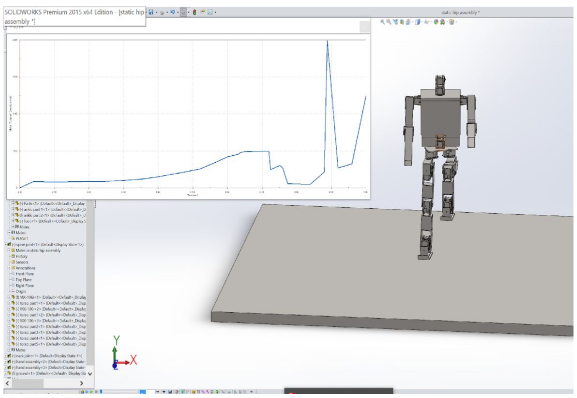
This is the torque of the right leg. It peaks at 5.5 N.m which we can be taken as the total torque on the motor when balanced on 1 foot. It then drops steeply as the bot falls down. For safety the other motor in the foot was checked and the value came out to be a mere 0.3 N.m.

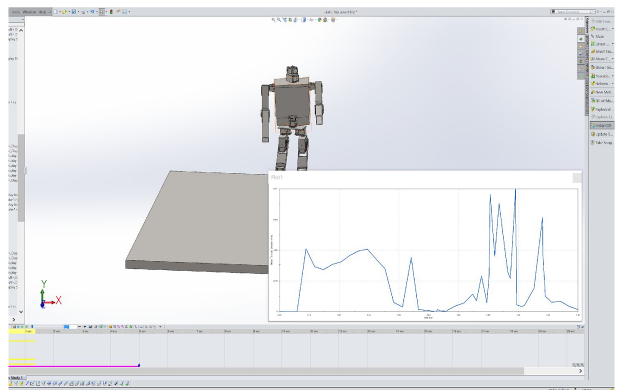


Case 3: Backward Falling

- Maximum torque in knee motor during falling backwards was 4.3 N.m.
- Maximum torque in spinal motor during falling backwards was 2.9 N.m.
- Maximum torque in thigh hip region was about 3.4 N.m.



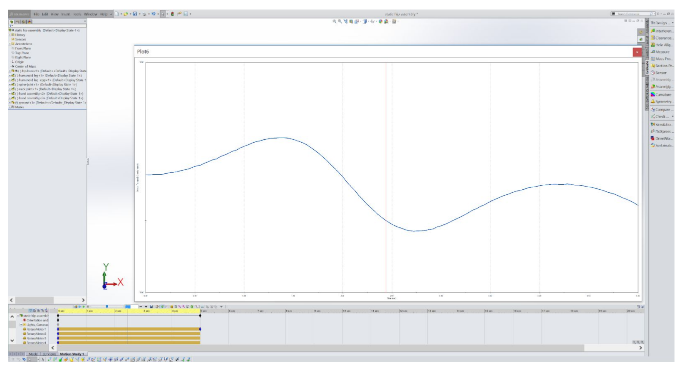




Case 4: Forward Falling

Fig Aside

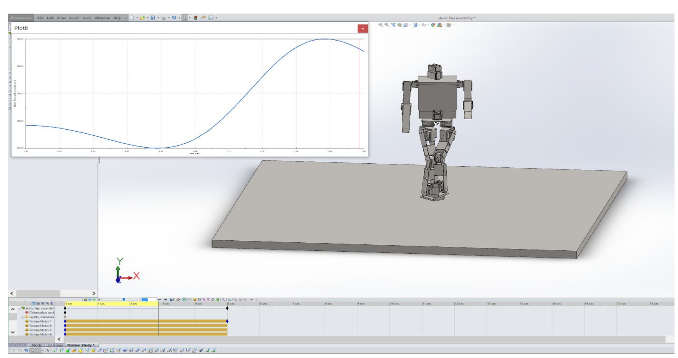
- Maximum torque in hip motor during forward falling was 6.7 N.m.
- Maximum torque in foot motor during forward falling was 5.5 N.m.
- Maximum torque in knee motor = 2.8 N.m



Peak torque of *ankle motor* was 5.096 N.m. The torque remained nearly constant i.e. between 5.095 N.m and 5.096 N.m.

Peak torque of *knee motor* was 3.303594 N.m. The torque remained nearly constant i.e. between 3.303574 to 3.303594 N.m.

The *hip motor* along pitch gave maximum torque i.e. 7.0635 N.m. The torque remained constant between 7.063-7.0635



N.m

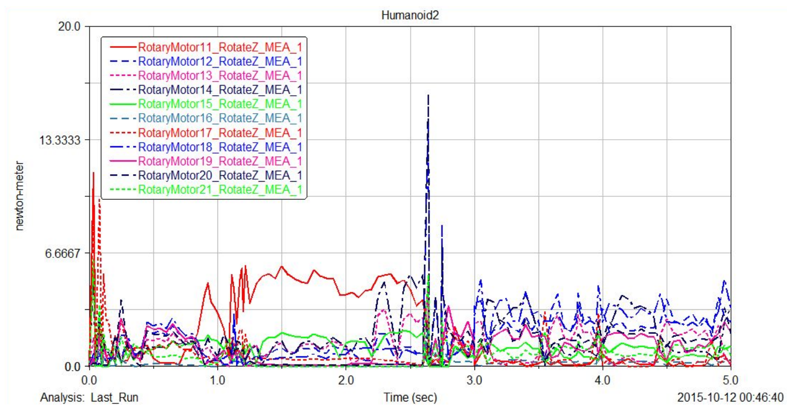
Motor Torques

Based upon the simulation studies we selected the appropriate motors which can withstand the peak torques experienced at the joints.

Simulation Torques (Simulation in ADAMS)

Table showing Maximum Torques experienced at various joints.

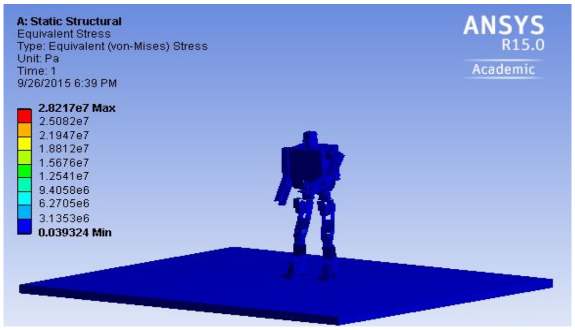
HIP (YAW)	2.8 Nm
HIP (PITCH)	5.5 Nm
HIP (ROLL)	6.7 Nm
KNEE (PITCH)	5.5 Nm
ANKLE (PITCH)	3.8 Nm
ANKLE (ROLL)	4.1Nm



Motor Selection based upon the calculated Torques

1. Dynamixel MX 106R (12 Nos.) ●Weight : 153g ●Dimension : 40.2mm x 65.1mm x 46mm ●Stall Torque ○ 8.0N.m (at 11.1V, 4.8A), ○ 8.4N.m (at 12V, 5.2A) ○ 10.0N.m (at 14.8V, 6.3A	**2. Dynamixel MX 64R (7 Nos.)** ●Weight : 126g ●Dimension : 40.2mm x 61.1mm x 41mm ●Stall Torque ○ 5.5N.m (at 11.1V, 3.9A), ○ 6.0N.m (at 12V, 4.1A) ○ 7.3N.m (at 14.8V, 5.2A)	**3. Dynamixel MX 28R (4 Nos.)** ●Weight : 72g ●Dimension : 35.6mm x 50.6mm x 35.5mm ●Stall Torque ○ 2.3N.m (at 11.1V, 1.3A), ○ 2.5N.m (at 12V, 1.4A) ○ 3.1N.m (at 14.8V, 1.7A)

4.Static Stress and Strain Analysis on ANSYS: Stress analysis was done in ansys to check the stress bearing capabilities and the structural stability of our model and as can be seen from the pictures the results were very good with a factor of safety of (FOS) = 4.25.



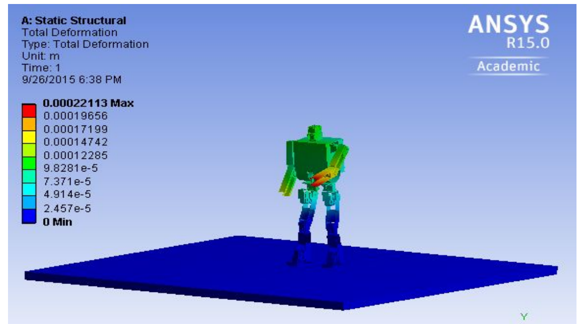
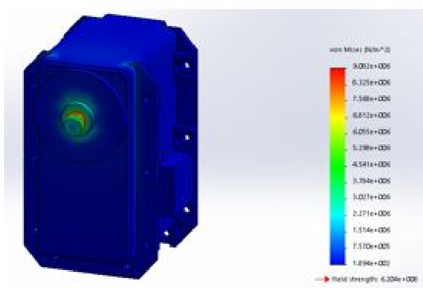


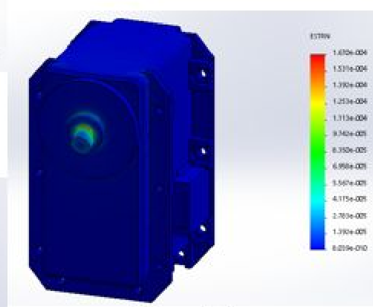
Fig 1: Stress Analysis *Fig 2: Strain Analysis*

We have also performed part-wise static force analysis-

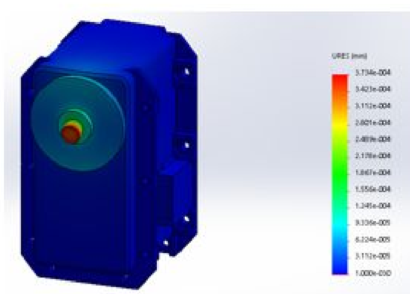
Static force analysis on motor



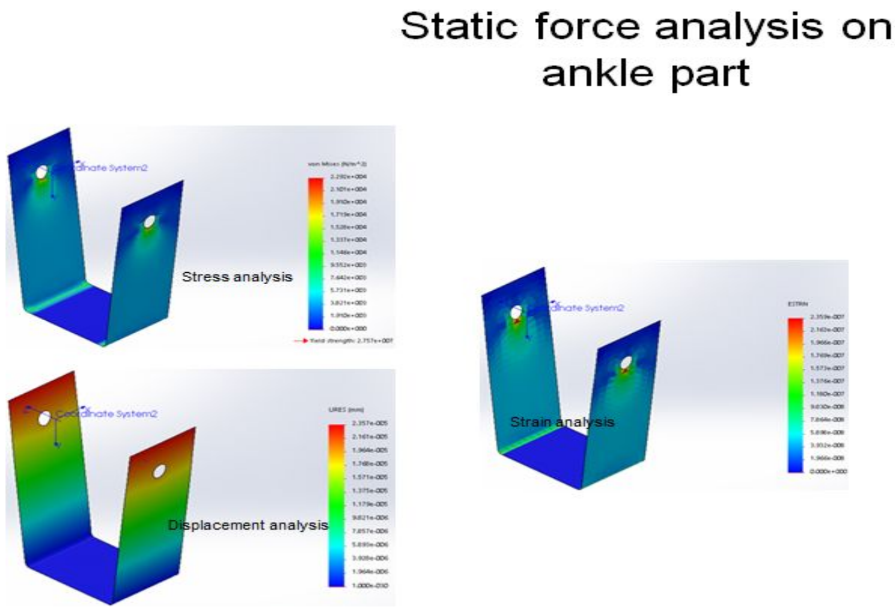
Stress analysis



Strain analysis



Displacement analysis

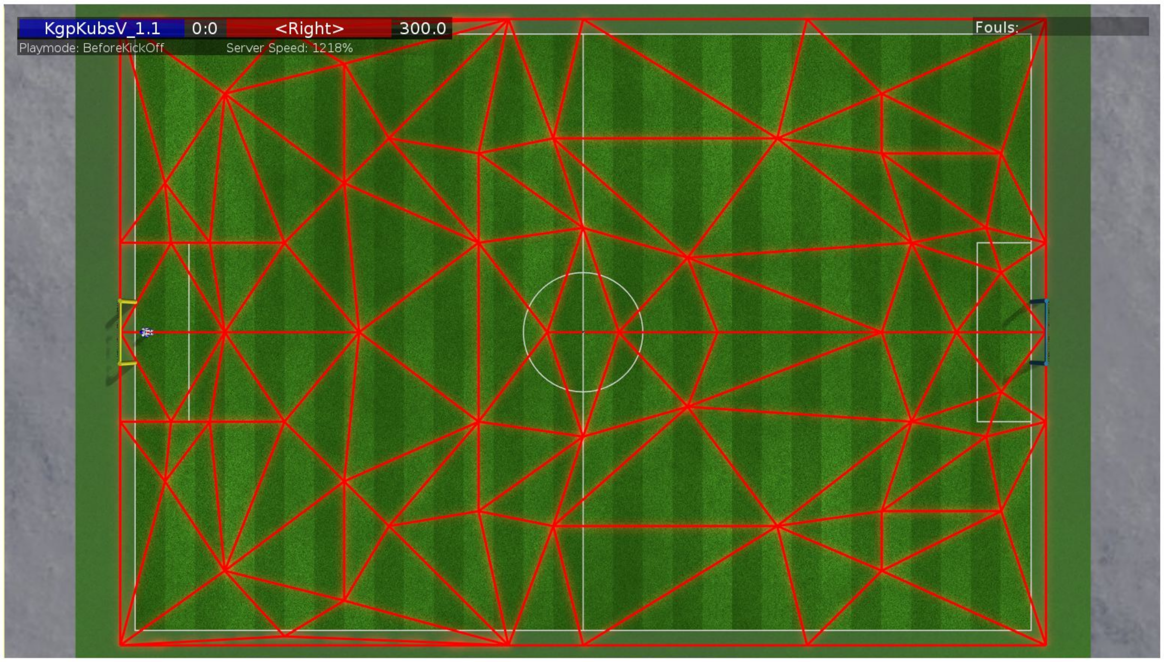


Our Approach to making a Humanoid

With the static analysis done for the humanoid, the prototyping of the humanoid will be done in two stages. First, we will work on 12 DOF biped that is capable of carrying a payload and is able to balance itself. This will basically serve as the lower body of the humanoid and it is the most challenging part of the humanoid. Once we are done with the dynamic simulation and shape optimisation of the biped, we will make a proper design of the biped ready to be manufactured. After this it will be put through rigorous testing and analysis.

Once we are satisfied with the results of the biped we will head on to the second stage that is developing the upper body of the humanoid. This will be followed by testing and analysis of the upper body. Then we will integrate the the upper and lower body and test it. Alongside all these the image processing, controls and playing strategies part will also be developed and tested.

Appendix X : Humanoid Control and Movement



Overview:
We have a decentralized system in which 11 humanoid bots need to perceive the environment data and act accordingly. A 11 vs 11 Humanoid Football match is run on a simulator and a monitor is connected to the simulator using UNIX sockets. Each Agent(Humanoid) connects to the simulator and runs its code on individual processes. Joint perceptors provide the agent with noise-free angular measurements every simulation cycle (20 ms), while joint effectors allow the agent to specify the speed and direction in which to move a joint. Visual information about the environment is given to an agent every third simulation cycle (60 ms) through noisy measurements of the distance and angle to objects within a restricted vision cone (120◦). Agents are also outfitted with noisy accelerometer and gyroscope preceptors, as well as force resistance perceptors on the sole of each foot. Additionally, agents can communicate with each other every other simulation cycle (40 ms) by sending 20 byte messages.

Code Base:

Major Modules:

- WorldModel:

It contains various environment variables which are updated by the data received by the humanoid. This include joint angles of all agents, ball position, agent positions, etc. Our work on this includes :

1. Implementation of **Roboviz** debugging tools which includes plotting of various points(Target positions of bots), drawing lines when necessary(for having a visual of the goalie stand line) etc.
2. We also implemented various heuristic based functions :
   a. Finding the best bot to attack the ball.
   b. Finding the nearest opponent(important parameters include FALLEN state of agent,direction in which agent is facing,distance from my position).

c. Finding the nearest teammate.

- **NaoBehaviour:**
  - *Walking:* We use a double linear inverted pendulum model omnidirectional walk engine. So, basically at the lowest level we have to send motor speed and angles but using a pre-built walk engine of AustinVilla we also have a set of parameters for walking. Optimizing this would get us a better walk speed. More about this in optimization section.
  - *Attacker:* The Attacker bot is arguably the most dynamic bot on the field.This role is selected based upon certain heuristics like fallen status,distance from opponent etc. The attacker can quickly dodge, dribble, kick(fast and slow) and drive ball to goal.
  - *Goalie:* The Goalie is the last line of defense and is the only agent allowed to purposely dive to try and stop a ball when the opposing team shoots on goal.Our goalie agent is designed to stay on a line .5 meters above its own goal line and always position itself between the ball and the goal so as to minimize the maximum angle between either goal post, ball, and the goalie. We apply **Kalman filter** to track the ball's position and velocity.We equip our goalie with a special set of diving skills including left/right dive , central stretch dive ..in order to effectively use its body to stop a ball that is headed toward the goal.
  - *Passing:* Passing is one of the most essential element for a multi-agent football playing system. We implemented a **fuzzy logic** based passing, which for each team member within a threshold radius takes into account.Following factors are taken into consideration while deciding the Pass.
    - target player's distance from source player
    - distances of opponent players from source player and target player
    - proximity to goal of source player and target player
    - angle to rotate for source player to face target player

    And based on these we compute a score, if the score is above some threshold we deem ourselves in a favourable position to pass, otherwise we continue as it is.
  - Dodge: The Agent plans a plan if an opponent is nearby such that he dodges the opponent.
- **Positioning:**
  - Delaunay Triangulation and Formations:

    Delaunay triangulation is the Dual graph of Voronoi cell plane. Hence, Delaunay triangles ensure that no other focal point lie inside the circumcircle of the Delaunay triangle formed. Also, due to this property it tends to avoid skinny triangles. As a result, interpolating any point inside the triangle yields to a smooth-gradient continuous equation in terms of the coordinates of the vertices of the triangle.

    The algorithm used to generate player positions uses statistical data (bot and ball positions under different conditions of the game) and generates a data set of agent positions with respect to certain ball positions. In all 65 ball positions in strategic locations were identified and triangulated using the incremental algorithm to generate Delaunay triangles . For each ball position we defined formation of agents. Once the triangles are generated, the **Gouraud Shading** algorithm yields the value of bot positions at any given point in terms of the values of bot positions stored at the vertices of the triangle enclosing it.
- **Role Mapping:**

- Hungarian Role Mapping: We use hungarian algorithm which solves the role assignment problem in polynomial time. Time complexity of this algorithm is O(n^3). At every one second,as per the ball position, we get a set of target Points from voronoi triangulation.These set of target points are then matched to players on field by hungarian algorithm. The cost function used for hungarian algorithm is euclidean distance between bot's current position and target location. It is also easy to see visualize that using this type of role matching has following properties:
  - (a) The collisions are mostly avoided.
  - (b) Longest distance is minimized
  - (c) It is dynamically consistent
- Prioritized Role Mapping:
  - The role map generated using hungarian is not always reliable for some crucial roles like attacker, defender and support attackers.
  - This roles are to be assigned based on game scenario rather than optimizing the distance travelled. We use heuristics to determine these roles. This gave a lot of advantage in the matches.

- **Optimization:**

  For optimizing low level skills, like walking, it may seems that reinforcement learning is more suitable but on contrary **CMA-ES** performs at par with the RL algorithms. We treat it as a black-box optimization algorithm, and have not interfered with the algorithm itself. For training a different cost function for optimizing different skills was used. For example, commanding the bot to go straight and using the difference in x-coordinate optimizes walking straight as well as improves its speed. Similarly other walk types (like lateral walking) and other skills like kicking could be optimized. Here we provide a brief about the CMA-Evolutionary Strategy:

  CMA-ES is a policy search algorithm that successively evaluates sets of candidates. Each candidate is evaluated with respect to a fitness measure. The next set of candidates is generated by sampling multivariate normal distribution that is biased toward directions of previously successful search steps. Recombination amounts to selecting a new mean value for the distribution. Mutation amounts to adding a random vector, a perturbation with zero mean. Adaptation of the covariance matrix amounts to learning a second order model of the underlying objective function. It is a parallel search algorithm so it can be run on a large server to make the optimization feasible. Some parameters were carefully chosen for optimization keeping others constant to reduce the search space.