Study of Formation Control of Mobile Robots

Shakir Mahmood, * Muwahida Liaquat, Zeeshan Ahmad, Ahsan Zaheer, and Abdullah Khan
Department of Electrical Engineering, College of Electrical and Mechanical Engineering,
National University of Sciences and Technology, Rawalpindi, Pakistan
Email: shakirmahmood1996@gmail.com, muwahida@ee.ceme.edu.pk, {zeeshan6803, ahsanzaheer123}@gmail.com, abdullah_khan485@outlook.com

Abstract **—Collective of Robots can perform complicated tasks that are beyond the capabilities of individual. The ability of such a collective to do tasks efficiently is because of its characteristic to form different shapes. In this paper, we present a Hybrid control method called Self-Assembly (SA), which will enables the collective to maintain the desired shape. This control mechanism will allow the robots to form different shapes with robustness and will provide the efficient techniques to counter Localization, Formation and Communication problems of the collective swarm.**

Index Terms **—formation control, localization, robot swarms, mobile robots, self-assembly, hybrid control**

I. Introduction

In Hybrid control method, we present a solution for the problem of formation and controlling collective of robots (also known as Swarm). The robotic collectives can do wonders which are beyond the capabilities of a single robot. One such wonder of a collective that gives it capabilities beyond that of the individual robots, is its shape. For example, in the collective, robots can form the shape of a wheel which will enable them to move faster in comparison to an individual [1]. In a swarm, consider a single robot has to cross the canyon to reach the goal by itself, the robot will not be able to cross the canyon but with the collective forming bridge, robot will be able to reach its goal. Control methods mostly used are decentralized but because of the formation of very limited number of shapes the requirement of initial seed is required to start the formation [2]. The ability of centralized control to form different generic shapes resulted in the formation of Hybrid Control. This scheme requires a “Seed” robot (the robot’s position is known and will act as leader for desired period) to start it and once the control scheme is in process the seed is no longer needed [3, 4].

In shape formation, each robot will move in order to assemble the desired shape. Robot should make sure that it moves in a space that is not previously occupied. One way to avoid blocking by neighbors is to use a gas model for robot movement [5]. This model uses the distance scheme i.e. robot will move away from neighbors if it is too close, thereby avoiding collisions. Some collectives [3, 6], cannot form shapes that have hollow spaces inside. Thus, if a robot gets stuck in undesired location for too long, random movement helps it get out [7, 8].

Shape formed by a collective must be robust to any damage. Control methods like [6, 8] cannot recover from the damage. Some species in nature like hydra can regenerate their body parts if cut into half but the new body is reformed at half of its original size [9]. Some methods [4, 7] keep the size of the shape same while the robots density may increase or decrease. In such case, there is an upper limit to the number of robots. Another issue is that how robot will know where they are located i.e. their current location with respect to his neighbors and the desired shape region. In most approaches, a coordinate system is used by the robots to form the desired shape which might be produced using trilateration [10], robust quadrilaterals [11] or MDS-MAP [12]. This help them in deciding their location with respect to the other robots as well as its distance from shape region. An initial seed starts its coordinates at (0, 0). Thus, when a robot comes close to the “Seed” robot whose location is already known, it determines its own distance and in turn will find its location in the coordinate system [10]. This approach is hybrid controller. In the work prescribed, we used the hybrid control method with almost no single point failure and the ability to form large class of shapes.

II. Proposed Method for Self-Assembly

A. Assumptions

The robots are homogeneous. Each robot is treated as a 2D circle with defined radius equal to Robot. It is capable of moving in XY-plane and rotating around its center. In a shape, a robot cannot share the place with another robot. All robots are identical and have their unique IDs. A whiteboard is used as a platform for their movement.

B. Method Description

The following section will describe the method for Self-Assembly, a Hybrid scheme to control the robots in collective. Each robot knows about all the information of desired shape. When Self-Assembly (SA) is implemented on each robot in the collective, the robots will self-assemble them to form desired shape.

C. Area of Operation

To generate a coordinate system, we proposed to divide the white board into cells. And area of each cell should be at least diameter (of the robot) square (D^2).

Manuscript received March 25, 2019; revised December 7, 2019.

doi: 10.18178/ijmerr.9.1.111-116

D. Robots Movement

Each robot will be placed properly in cells on the white board. Robot will move a distance of at least one diameter, for every movement it makes. This will ensure that robot is always completely in the cell. This will be helpful to determine its location and relate it to the image which is discussed next.

E. Image Formation and Conversion

As mentioned earlier, we used Adobe Illustrator to make image of the desired shape. Some of the examples are shown in the Fig. 1. One limitation is that the image should be made with two primary colors red and green. Where the red color represents the shape region and green region represents anti-shape region. The image can be of any size (1920x1280, 1280x1280, 1280x720, 640x320 etc.) The red and green planes of the image are then normalized to one. Thus, in red plane red pixels are on (intensities = 1) pixels and green pixels are off pixels (intensities = 0) whereas in green plane red pixels are off pixels and green pixels are on pixels. Later these pixels will determine whether robot is in shape region or anti-shape region.

It is fed to every robot and is then converted to the desired dimensions. The advantage of this conversion is that it can create any image and feed it to the robots. The conversion of image is done using the concept of contrast stretching. The size of the converted image depends on the number of cells on the white board.

Each pixel of the image represents a cell. For example, if there are 10 rows and 12 columns of cells on white board then the dimensions of the converted image will be 10×12.

In contrast stretching, we improve the contrast in the image by stretching the intensities of the pixels to span over the desired range. It is done using simple equation of line as shown in (1)

$$y - y1 = m(x - x1) \quad (1)$$

where m is the slope of the line given as

$$m = \frac{y2 - y1}{x2 - x1} \quad (2)$$

In terms of contrast stretching this equation is written as in (3)

$$T(x) = (x - x_{min})\left[\frac{L-1}{x_{max} - x_{min}}\right] \quad (3)$$

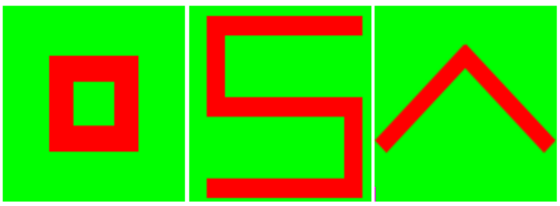

Figure 1. Desired shapes

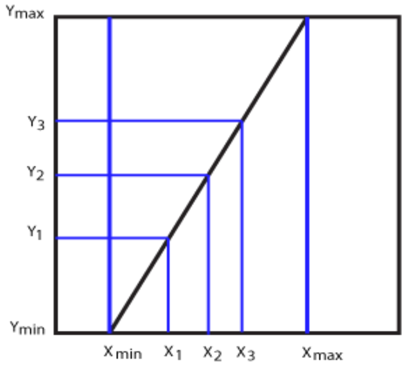

Figure 2. Range stretching

Here L is the total number of grayscale levels. It is demonstrated in the Fig. 2.

We used this concept to convert the dimensions of any image to any desired dimensions. For that, we have to use 2D form of this equation as an image is sort of a 2D plan with some width and some height. The equation for image size conversion is as follows

$$y = \left(\left(\frac{x - x_{min}}{x_{max} - x_{min}}\right)(y_{max} - y_{min})\right) + y_{min} \quad (4)$$

$$[X\ Y] = \left(\left(\frac{[x'y'] - [1\ 1]}{[m\ n] - [1\ 1]}\right)([M\ N] - [1\ 1])\right) + [1\ 1] \quad (5)$$

where $[X\ Y]$ are the coordinates of the input image. $[x'\ y']$ are the coordinates of the desired image. $[m\ n]$ are the maximum dimensions of the desired output image. $[M\ N]$ are the maximum dimensions of the input image (made with Adobe Illustrator). [1 1] is the minimum dimension of both the images.

For any coordinates $[x'\ y']$ of the output desired image we find the corresponding coordinates of the input image $[X\ Y]$. The intensities of the pixels of input image at those coordinates are then assigned to the corresponding pixels of the output image (Output $(x'\ y')$ = Input (X, Y)). Thus, by selecting m and n, we can map image of any size to any desired size.

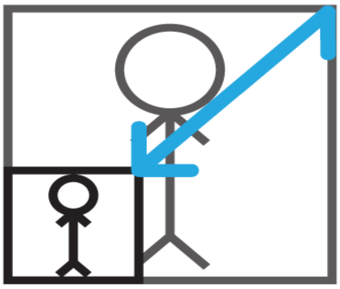

Figure 3. Image conversion

F. Localization

All the robots have the image and every robot knows the locations of 'on' and 'off' pixels. So, if the robots know their own locations they will simply calculate their distance from each pixel of the shape region and move towards it. Trilateration is the most suitable technique for localization of robots. This technique depends on the distances from and coordinates of at least three points. The robots have an IR sensor for communication which helps in determining the distance between the sender and receiver. Strength of the received signal helps to determine the distance. But there is a limit to the range of communication.

To find the location of an unknown point, three known points and distance of the unknown point from the three known points is required.

1) The unknown point is (x, y)
2) Three known points are $(x1, y1)$, $(x2, y2)$ and $(x3, y3)$

Distance of unknown point from the three points are

$$d1^2 = (x - x1)^2 + (y - y1)^2 \quad (6)$$

$$d2^2 = (x - x2)^2 + (y - y2)^2 \quad (7)$$

$$d3^2 = (x - x3)^2 + (y - y3)^2 \quad (8)$$

Subtracting equation (6) and (7) gives us

$$d1^2 - d2^2 - x1^2 + x2^2 - y1^2 + y2^2 = 2x(x2 - x1) - 2y(y1 - y2) \quad (9)$$

$$d2^2 - d3^2 - x2^2 + x3^2 - y2^2 + y3^2 = 2x(x3 - x2) - 2y(y2 - y3) \quad (10)$$

This gives us two equations in terms of two unknowns. Thus, these can be solved simultaneously to find the unknown coordinates (x, y).

III. Experimental Verification

To verify that the entire task of forming different shapes and SA is achieved, we tested it on simulated number of robots. A 10 × 10 axis was chosen where a total of eight robots were randomly distributed. In each simulated time step, the controller would run once, which will command a movement for the robot. And at the end the robots could make the necessary movement change and communicate with their neighbors by sending messages.

Firstly, an image of the shape is selected which the robots need to follow. The image is the shape which would be followed by each robot and occupy their positions in the shape region. The image is converted into the desired coordinates and it contains both the regions i.e. shape region and anti-shape region. The image of any size is converted into 10 × 10 size which represents 10 rows and 10 columns. Thus, in the shape region the shape region pixels are on and in the anti-shape region the anti-shape pixels are on. This helps the robots know that in which portion or part of the shape they are lying. The image is fed to every robot and is then converted to the desired dimensions. The size of the image will depend on the number of cells total. Whereas each pixel of the image represents a cell.

The image conversion is done using contrast stretching. By using contrast stretching the contrast in the image is improved by stretching the intensities of the pixels to span over the desired range. The concept converts the dimensions of any given shape to the desired dimensions.

Three of the robots have already been assigned positions in the 10 × 10 axis. This will help to get other robots localized with the help of technique called ‘Trilateration’. The seed robots occupy fixed positions whereas the remaining five robots tend to move randomly until they come into the vicinity of the three seed robots. The three robots know their absolute positions. They are distributed at random positions. The seed robots will constantly be transmitting their coordinates and with the strength of the received signal, the robots in the vicinity of these seed robots will determine the distances and will know their coordinates.

Once they come into the vicinity of the seed robots they get themselves localized by applying the equations discussed in Section II. If there are three or more than three neighbors of a robot then localization function locates the non-seed robots. Similarly, all the robots move randomly and get themselves localized when they come into the vicinity of the seed robots.

Once a robot gets localized it becomes a seed robot itself. It is necessary because there is a possibility that seed robots get surrounded by a crowd of robots. Robots would be requiring absolute positions to estimate distance between them. The absolute positions will be required until all the robots become localized. The speed of the operation would also increase since there would be more number of seed robots and less number of unlocalized robots.

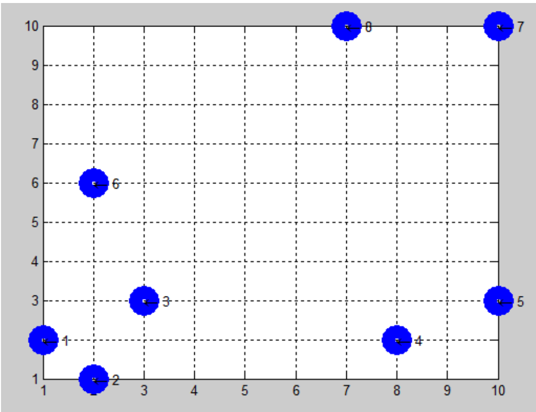

Figure 4. Seed robots and localization

Once all the robots are localized, they are required to move in the shape region. Robots know their own position and the coordinates of the shape region. So, robot will find the distance from all the coordinates of the shape region and then it will move towards the cell that is closest to it.

To ensure the smooth movement of robots, it would use the angle along which the robot will move along a straight path to reach the end point. Whereas the angle is found by

$$\theta = \tan^{-1}\frac{y_2 - y_1}{x_2 - x_1} \tag{11}$$

The robots which are already in the shape region need not to be moved, whereas those robots which are outside the shape region, need to be moved inside the shape region and in this way their random movement helps them calculate the minimum distance from the shape region. While moving randomly the robots, after each movement check the state and see whether they have entered in the shape region. If the cell present in the shape region is already occupied by a robot, the robot would again move randomly and then find the distance to the cell which is the next closest to it. In this way all the robots will eventually get in the shape region. Thus, making the desired formation.

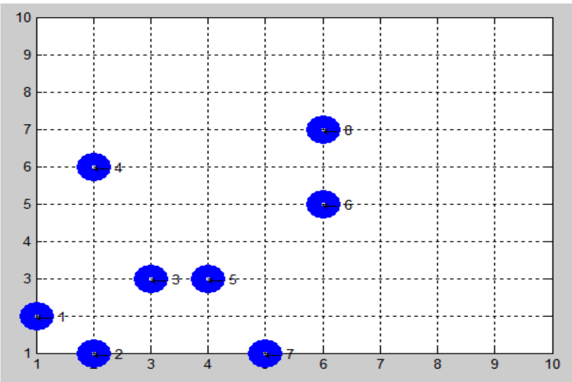

Figure 5. Random movement of robots which will help them move into the shape region.

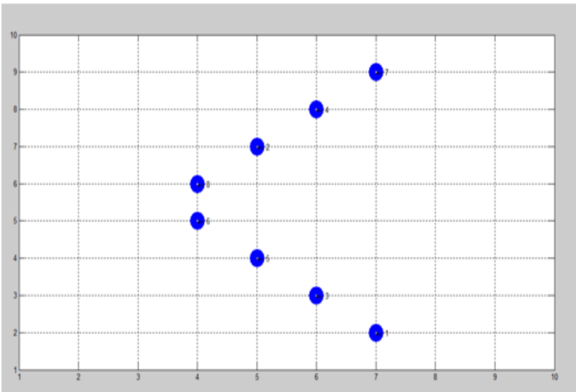

Figure 6. Shape Formation after robots enter into the shape region.

Once all the robots have been moved into the shape region occupying different positions, the desired formation is achieved, and all the robots are occupying positions and their coordinates are inside the shape region.

A. Limitation

One limitation with this method is that the three known points must be non-collinear. If these points are collinear then above equations give indefinite solution. Let us explain this with an example

Suppose we have three points $(0,1)$, $(0,2)$, $(0,3)$. What happens here is on the right side of equation (9) and (10) the term with "x" becomes zero. Thus, solving these equations gives indefinite results.

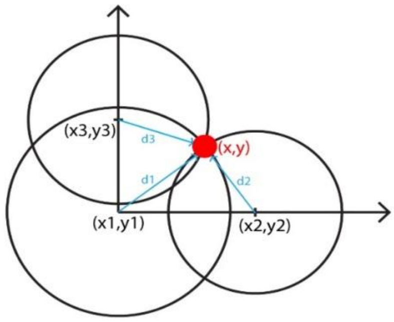

Figure 7. Trilateration

IV. Choice of Robots

Depending upon the requirements of proposed algorithm, Kilobots were the best option as they are simple and cheap and can move in any direction, communicate and measure distances from each other.

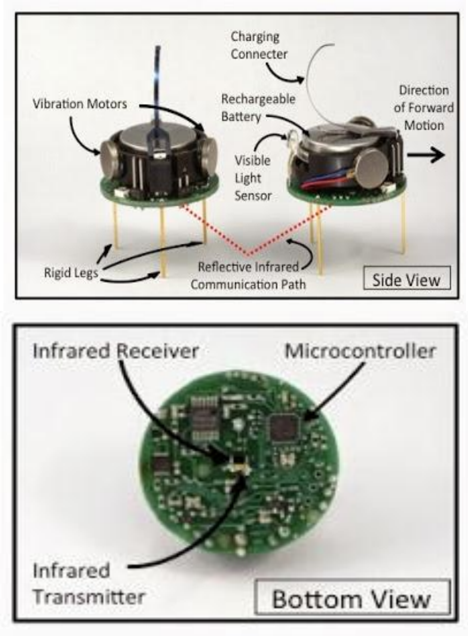

Figure 8. Kilobot components

First step is to select the seed robots. We randomly chose three robots as seed robots and placed them in each other's vicinity. The non-seed robots will move randomly until they come into the vicinity of seed robots. Thus, using trilateration the non-seed robot will get localized and become the seed itself. In this way, all the robots will get localized. The robots have an image which contains the coordinates of the shape region and the robots know their own positions. Every robot will calculate the distance from the coordinates of the shape region and will move towards the one with least distance but if some robot is already present at that coordinates, the moving robot will again measure distances from shape region coordinates and move towards the one that is second nearest. This is a simple method that can ensure that the robots form the desired shape.

A. Results

We made a line shape on Kilobots. Here are the results.

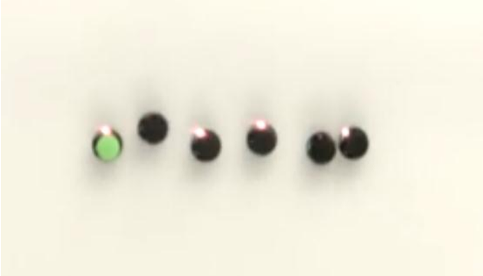

Figure 9. Line Formation

V. Conclusion

The problem of swarming among mobile robots poses different risks caused by the physical limitations of robots. There are a limited number of algorithms available on the concept of swarming, most of which face threats due to physical limitations of different robots which includes aspects like computational power, sensing capabilities, number of sensors on board etc. We were able to develop an algorithm which can be implemented on a robot as cheap as Kilobot having only one IR sensor at its disposal. Verification of our finding was done on MATLAB which proves the working of our algorithm on software defined robots imitating real robots as closely as possible.

We were able to apply the concept of localization on the bots and constructing a line formation was also done using proposed algorithm.

VI. Future Works

Robots can enabled to learn from their surroundings and make their own decisions. For example, if there are to traverse through a terrain full of obstacles while maintaining their formation, they would first move through that terrain randomly, keeping records of the obstacles and elevations or depressions in that terrain and when they have enough information they would start making the formation and move through the terrain with the minimal changes to the formation. This formation control and traversing can open many new gateways to the military operations as well as for explorations of some dangerous sites where neither human beings nor a single robot can survive.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

This work was carried as final year project of Shakir Mahmood, Ahsan Zaheer, Zeeshan Ahmad and Abdullah Khan under the supervision of Dr. Muwahida Liaquat. This Research was funded by National University of Sciences and Technology, Pakistan.

Acknowledgment

Thanks to National University of Sciences and Technology for sponsoring this research.

References

[1] H. Chiu, M. Rubenstein, and W. Shen, "Deformable wheel"-A Self-recovering Modular Rolling Track," *Intl. Symp. Distributed Robotic Systems*, 2009.

[2] E. Sahin and T. H. Labella, et al., "SWARM-BOT: Pattern formation in a swarm of self-assembling mobile robots," *IEEE Int. Conf. Systems, Man and Cybernetics*, Hammamet, Tunisia, 2002, pp. 145-150.

[3] K. Stoy and R. Nagpal, "Self-repair through scale independent self-reconfiguration," *Int. Conf. on Intelligent Robots and Systems*, Sendai, Japan, 2004.

[4] D. Arbuckle, "Self-assembly and self-repair by robot swarms," dissertation, University of Southern California, Aug, 2007.

[5] D. Arbuckle, and R. Aristides, "Active self-assembly," *Int. Conf. on Robotics and Automation*, New Orleans, LA, 2004.

[6] M. Yim, J. Lamping, E. Mao, and G. Chase, "Rhombic dodecahedron shape for self-assembling robots," Xerox PARC SPL, Tech. Rep. P9710777, 1997.

[7] V. Zykov and H. Lipson. "Experiment design for stochastic three dimensional reconfiguration of modular robots," *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, 2007.

[8] J. Bishop, et al., "Programmable parts: a demonstration of the grammatical approach to self-organization," *Int. Conf. on Intelligent Robots and Systems*, 2005.

[9] H. Bode, "Head regeneration in a Hydra," *Developmental Dynamics*, vol. 226, pp. 225-236, 2003.

[10] P. Maxim, S. Hettiarachchi, W. Spears, et al., "Trilateration localization for multi-robot teams," in *Proc. 6th Int. Conf. on Informatics in Control, Automation and Robotics*, 2008.

[11] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements," *SenSys*, 2004.

[12] Y. Shang, W. Ruml, Y. Zhang, and M. Fromherz, "Localization connectivity in sensor networks," *IEEE Trans. on Parallel and Distributed Systems*, vol. 15, Oct. 2004.

Mr. Shakir Mahmood was born on 4th April, 1996 in Rawalpindi, Pakistan. Always wanting to pursue a research oriented career, he graduated in 2018 with an Electrical engineering degree from National University of Sciences and Technology, Rawalpindi, Pakistan. During his bachelors, he worked on many robotics based project and started research with his colleagues on the swarm robotics.

In the last summers of his bachelor's degree, he worked with an Internet of things (IoT) based company and gained a hands on experience of IoT while working on the ongoing projects. One of those projects, an IoT development board, was later launched as a product currently used by students of many universities in Pakistan. He intends to use to the same experience gained for the robotics. His areas of interests included Robotics, Control Systems, Artificial Intelligence and Internet of Things,

Muwahida Liaquat received her BE in computer engineering and MS in Electrical Engineering degrees in 2004 and 2006 respectively from National University of Sciences and Technology (NUST) Pakistan. She did her PhD in Electrical Engineering from NUST in 2013 with specialization in control system. Her work was focused on developing a reconstruction filter for the sampled data regulation of linear systems. She is currently working as an **Assistant Professor** in the Department of Electrical Engineering at College of Electrical and Mechanical Engineering which is a constituent college of NUST, where she is part of the control systems research group along with teaching specialized courses in control engineering to both graduate and post graduate levels. She has also supervised a number of MS students in the area of control engineering. Her areas of interests include Networked Control Systems, Multi-rate and Sampled-Data Systems, Control of Impulsive Systems and Optimization, Image Processing and Industrial Automation.

Mr. Ahsan Zaheer was born on 24th of August, 1995, in Rawalpindi, Pakistan. He completed his bachelor's in Electrical engineering from National University of Sciences and Technology, Islamabad.

In 2017, he worked in Attock Gen limited, an Oil and Gas company, where he worked on thermal power plant and gained experience on power generation, distribution, and different kinds of generators. He is currently employed in Schulumberger as **Wireline Production Services engineer**.

Throughout his bachelors, he was among the high achievers and achieved many pointer badges. His major areas of focus are power system analysis and robotics.

Mr. Zeeshan Ahmad was born on 24th of April, 1997 in Nowshera, KPK, Pakistan. He has completed his preliminary education from Wah Cantt. For his higher studies, he pursued Bachelor's in Electrical engineering from National University of Sciences and Technology (NUST), Islamabad, Pakistan. He was part of the team that researched on the formation control of mobile robots and is on the lookout for the opportunities in research and development.

Mr. Ahmad worked as a Vice-Chair person for IEEE, Campus Branch and has also held the student membership for the session 2017-18.

Mr. Abdullah Khan was born on 30th March, 1996, in Wah, a village near the capital of Pakistan, Islamabad. He earned his bachelor's degree in Electrical engineering in 2018 from National University of Sciences and Technology, Rawalpindi. His majors are Electrical Power engineering and Robotics which include the study on Renewable Energy Sources, Power System analysis and Kinematics of different robots.

He attended an internship program at Pakistan Ordinance factories (POF), Wah cantonment. The major areas of the internship were the Power house, understanding of the SCADA architecture system, getting to know about the weapons factory and the manufacturing, generation of the Thermal power which is used to power the area.

Mr. Khan has been working dedicatedly throughout his educational career and has also made an impact on some of the big organizations like POF. He was among the high achievers of his university and has been awarded with pointer badges.