ROBOTICS: WHAT, WHY, HOW AND FURTHER

K J SARMA, SMIEEE
MALLA REDDY ENGINEERIGN COLLEGE (A),
Dhulapally, Maisammaguda, SecunderabD-500 100.

Abstract: We see clearly automation is taking over the clean production, communications and services In many instances for the purposes of good quality, reliability, control, precision and the related measurements robots are used. In this 21 st century the manufacturing industry in India is also moving into the use of robotics. As India is moving towards make in India, made in India and digital India, it is necessary to induce an enthusiasm and encourage youngsters to keep aware of the developments, Challenges and initiate research activity on Robotics. This paper presents an overview of the possibilities existing for theoretical research on robotics.

Keywords -- *Robotics, classification, types, research, algorithms, Mathematical and statistical models, research trends.*

I. INRODUCTION:

Robots are the artificial devices that replicate the human work, while robotics is an art of how to develop a robot. Hence a machine, resembling with human being and is capable of executing actions automatically; especially programmable by computer is a robot. Thus human actions have to be programmed for the machine to execute.

While writing a play, Karel Čapek struggled to come up with a word to name the robots, initially settling on 'laboři', from the Latin 'labor'. He discussed this with his brother, Josef, and Josef suggested 'roboti', which gave rise to the English 'robot'. 'Roboti' derives from the Old Church Slavanic 'rabota', meaning 'servitude', which in turn comes from 'rabu', meaning 'slave'. The full title of translating into English as *Rossum's Universal Robots,* which debuted in January of 1921[8,9]

ROBOT can function only in its own special environment. Before we can understand the ROBOT or tell it what to do, we must first have a clear picture of the ROBOT's domain, or environment in which it operates. Most of the ROBOT's movements and capabilities are closely related to the characteristics of its domain.

To "program the ROBOT" means to devise a sequence of instructions designed to accomplish some particular task. To know what tasks are reasonable for the ROBOT, we need to know the ROBOT's characteristics and capabilities. Once we know what individual actions the ROBOT can perform, then we can speculate on what types of tasks it is capable of doing.

In general, robots are classified based on their capabilities. Some standard classifications of robots include their domain of operation, degree of autonomy, and the goal they are designed to fulfill.

Domain of Operation: Robots can be designed and built for any environment imaginable. One popular way of classifying robots is by what conditions and environment they're designed to operate in. Some typical examples include:

Stationary: These robots are fixed in one place and cannot move. This category includes robotic arms, computerized machine tools, and most other Industrial Robots.

Industrial Robots are robots used in mass production e.g. welding robots, CNC plate cutters or CNC drills. The large majority of these robots are stationary and tethered to a computer.

Ground: These robots are designed to operate on the surface of the earth or other planet, and are usually sub categorized by their drive train using wheels, tracks, legs.

Underwater: Also known as Autonomous Underwater Vehicles, these are designed to operate underwater, possibly at great depth.

Aerial: Unmanned Aerial Vehicles are various kinds of robotic flying machines, including planes and helicopters.

ISSN: 2278 – 7798

Microgravity: Robots that have been designed to operate in low-gravity environments, such as earth orbit, other specific hazardous environments[20].

There are many variations of Japanese robotics. Some different types of robots are: Humanoid Entertainment Robots, Androids, Animal (four legged) Robots, Social Robots, Guard Robots, and many more.

II. CASSIFICATION AND TYPES:

A classification is a means of placing a particular robot into a broad category or group where it can be compared with like robots in the same group for suitability for a particular application. A specification is a factual description of the robots size, capacity, capability, geometry etc. It is possible for a robot to fall within many systems of classification depending on the purpose for which it is being considered.

One common means of robot classification are as follows: Industrial robots which are grouped according to their structure or design, this is known as their configuration. Important considerations when selecting a robot concern its positioning accuracy, its repeatability and its ability to traverse smooth and often complex contours. These considerations are a function of the type of control system employed. Industrial robots may be powered by hydraulic, pneumatic, electrical and mechanical devices e.g. a high power lifting function will dictate a hydraulically powered robot to be specified. Industrial robots may be classified according to the particular task for which they are meant for.

Robotics in contrast to other branches is a reasonably new domain called multi-disciplinary domain of engg. The different branches occupies in the development of robotics are *mechanical engineering* – dealing with machinery and structure of the robots e*lectrical engineering* dealing with the controlling and intelligence by sense of robots; and *computer engineering* – dealing with movement development and observation of robots.

Another classification is due to the circuits of the robots and the variety of applications, it can perform. Further they are classified into Simple level robots (which do not contain circuit Ex. Washing Machines), Middle level robots (based on circuits and perform multiple tasks and are fully automatic like programmable washing machines) and Complex level Robots (which can be programmed and reprogrammed, contain sensors and contain complex circuits ex. Laptop / computer).

A third type are classified based on the type of distinctiveness which transforms depending on the atmosphere it works like outer space, intelligent home exploration, Military robots, farms, the car industry, hospitals, disaster areas, and entertainment.

A fourth type based on applications which vary from industrial robots, domestic or household robots, medical robots, service robots, military robots, entertainment robots, space robots, hobby and competition robots, etc.

Degree of Autonomy: Autonomy is the quality of being self-controlled. One measure of autonomy is the amount of human control that is required for the robot's operation. An autonomous robot can operate properly without intervention indefinitely and can deal with unexpected problems gracefully. Tele-operated robots constantly require humans to send the robot control signals. These are only the endpoints; there is a continuum of possibilities between them.

Robots can also be classified by how self-contained nature they have. Power, logic circuitry, and other things may be located either on the main chassis or connected via a cable tether or wireless link from another location.

Goal: Robots are also routinely categorized by the goals they are designed to achieve. These include contests, personal enrichment, manufacturing, and entertainment. The ROBOT's capabilities are defined by its hardware. The ROBOT has no eyes or cameras built in; therefore, it cannot "see" its surroundings. It has no numerical capacity. We cannot use it to perform a task that is beyond its capabilities. The ROBOT has no intelligence. It cannot think or plan its own activities unless programmed by intelligent models. The ROBOT is capable only of following a program of instructions that it retrieves from its memory and then executes one instruction at a time.

The ROBOT has a **memory** where in the stored instructions make up a program. In addition the ROBOT is equipped with circuits, which enable it to **fetch** instructions from its memory, interpret or **decode** their meaning, and **execute** them [23].

ISSN: 2278 – 7798

Memory - The ROBOT's built-in memory is quite tiny. It contains 32 memory locations, numbered consecutively from 0 to 31. Each memory location is capable of storing a single ROBOT instruction.

Instructions - The ROBOT does not understand human speech or any written language. It responds only to binary numbers, each eight bits (binary digits) in length. Each possible combination of the eight digits is a meaningful instruction to the ROBOT.

Fetch - The ROBOT's electronic circuits cause it to fetch (or retrieve) its instructions from memory, one at a time, and usually in the order in which they are stored. The only exception to this rule involves the GOTO command from the ROBOT's instruction set. For the full story, see the explanations of the individual ROBOT commands and the sample programs.

Decode - The ROBOT knows what action to carry out for a given binary number because the ROBOT's hardware contains an instruction decoder. Such a unit is present in one form or another in all digital computers. The instruction decoder is a set of circuits that causes the appropriate actions to be taken based on the particular binary number instruction that is received as input. In most cases, the decoder circuits may be required to split apart a given binary number into two or more parts in order to interpret the instruction represented by the number. Each ROBOT instruction must be split into two segments. The first is the operation code or command part of each instruction is represented by the first three binary digits, starting from the left. The remaining five bits are ignored by the ROBOT except when used with the GOTO command.

Execute - The ROBOT executes the instructions by carrying out the action or command that each individual instruction represents. Each instruction has its own particular purpose and characteristics. Some instructions cause the ROBOT to take an observable action. For example, 00000000 causes the ROBOT to take one step forward. Instructions of this type may cause different results depending upon where the ROBOT is located; this is what makes programming the ROBOT both interesting and a little tricky. For example, if the ROBOT is directly in front of a wall, the 000000 instruction causes it to spin its wheels and wear them out, much to the detriment of the ROBOT and the wall.

Software - The Robot's Language: The ROBOT's language consists of eight different commands. These are referred to as the ROBOT's instruction set. Every different type of computer has its own instruction set.

Each ROBOT instruction consists of an eight-bit binary number. Up to 32 ROBOT instructions may be stored in the ROBOT's memory, one in each memory location. Thus, the maximum number of **statement**s in any one ROBOT program is 32.

The list of instructions, which the ROBOT has stored in its memory, can be determined and changed by the person who operates the ROBOT. The particular list of instructions that is currently stored is referred to as the ROBOT's program, and it must be placed into the ROBOT's memory before any execution can take place. To place programs into the ROBOT, you must have some sort of access to the ROBOT's memory.

The process of setting the value in each memory location to conform to a particular program is referred to as **loading** the program. All computers must load programs before they can execute commands. When all the instructions of a program are loaded, the programmer close the door to the memory bay and presses the ROBOT's START button. The ROBOT can then begin to execute the program.

As noted earlier, there are exactly 32 memory positions. They are numbered from 0 to 31, using the binary numerals 00000 to 11111. The memory position numbers in the ROBOT's memory back start in the upper-left corner and are numbered down the first column and then onto the second column. When a program is executed, the instructions are performed in the order of step number, starting with the instruction stored at location 00000.

One of the ROBOT commands - the GOTO command - is designed to allow the ROBOT to take instructions out of normal order. This command or op-code has the binary code 101. When used within an instruction, its operand (the last five bits of the instruction) indicates the step number (memory location) from which the next instruction is to be taken.

For each value possible there is a three-bit op-code of an instruction along with a single English word that suggests the action taken by the ROBOT command. Opposite to each op-code is an explanation of the possible actions that can result from using that command in an instruction. We l

ISSN: 2278 – 7798

refer such set of commands as the ROBOT's instruction set.
Finally, the ROBOT is equipped with a special red warning light on top of its head. When the ROBOT's power switch is activated, the warning light is off, but certain situations cause it to be turned on. When the warning light is on, the ROBOT ignores all instructions except one, the one containing the LIGHT command. This feature gives the ROBOT a limited decision-making capability. These concepts follow, Program, Problem, Solution, Algorithm, Conditions, Loops, Infinite loops, Escape from loops. A **problem** in the context of this discussion is a well-defined task to be performed by the computer.

III. ROLE OF ALGORITHMS:

An **algorithm** is a step-by-step process used to solve a problem. Essentially, the algorithm is the solution to the problem and is usually implemented by a program.. A program is a sequence of instructions designed to carry out a task. A potential task for the ROBOT is a problem. The solution to the problem is provided by the program. But to arrive at the program, we must proceed through fairly detailed logical thinking, and devise a general method for approaching at the problem. Such a method, expressed in clear and precise logical steps is called an algorithm. This term is one of the words most commonly used by computer scientists and programmers[2,4,6,20].
As modern robots address real-world problems in dynamic, unstructured, and open environments, novel challenges arise in the areas of robot control algorithms and motion planning. These issues stem from an increased need for autonomy and flexibility in robot motion and working with the robot. Adequate algorithms have to be developed for control and motion planning for high-level motion strategies, that adapt to sensor feedback. Researchers of Algorithms for planning and Control of Robot Motion promote application-driven, towards these objectives. The priority areas include -- consideration of sensing modalities and uncertainty in planning and control algorithms; development of representations and motion strategies capable of incorporating feedback signals; motion subject to constraints, arising from kinematics, dynamics, and non-holonomic systems; addressing the characteristics of dynamic environments; developing control and planning algorithms for hybrid systems; understanding the complexity of algorithmic problems in control and motion planning; and encouraging the application of planning algorithms in novel application areas.

Robot algorithms are a fundamental component of robotic systems. These algorithms process inputs from sensors that provide noisy and partial data, build geometric and physical models of the world, plan high-level and low-level actions at different time horizons, and execute these actions on actuators with limited precision. The design and analysis of robot algorithms raise a unique combination of questions from many fields, including control theory, computational geometry and topology, geometrical and physical modeling, reasoning under uncertainty, probabilistic algorithms, game theory, and theoretical computer science. More specifically the algorithms are classified as

1. **Complexity Analysis of Algorithms (brief introduction/review):** What is an algorithm? Is an algorithm that you have developed efficient and correct?
2. **Discrete planning:** Many robotic tasks can be represented using discrete abstractions, e.g., graphs. How can we efficiently search a graph to find a desired goal state (node)?
3. **Workspace and configuration space:** What are the workspace and configuration spaces of a robot? How can we create discrete abstractions and cell decompositions for motion planning, sensor placement, coverage etc?
4. **Continuous planning (using sampling based methods):** How do we move a robot with complex kinematic and dynamic constraints in an environment with obstacles? What is the theoretical complexity of motion planning?
5. **Sensing and localization:** How do we model uncertainty in sensing? How does the robot know where it is?
6. **Planning under uncertainty:** Given the sensor inaccuracies and localization uncertainty, how do we plan the actions of our robots?
7. **Topics in algorithmic robotics:** (a) temporal logic motion planning and re-planning, (b) multi-robot issues: decentralized control, consensus, etc, (c) warehouse robotics and automation, (d) software quality assurance

ISSN: 2278 – 7798

in robotics, (e) other topics based on the research interests of the students attending the class.

IV. MATHEMATICAL DISCIPLINES FOR ROBOTICS:

Many mathematical disciplines are likely to have strong relevance for robotics. In what follows we have listed several mathematical disciplines, some cases and the potential application areas or subtopics [19,20]

1**. Algebraic and differential topology.** This work includes the application of techniques from loop space theory and low dimensional topology to understand configuration spaces of many-particle & many-body systems. Some techniques from global differential topology to analyze configuration spaces of arbitrary length closed chains in two and three dimensions with spherical, revolute, and prismatic joints are also used in the study of robotics. The tools are also directly relevant to - Distributed sensing and actuation systems, such as those made possible with MEMS and nanotechnology; and High dimension design problems, such as intelligent vehicle-highway and air-traffic management systems.

2**. Dynamical systems theory.** There has already been a fruitful interaction between the dynamical systems community and robotics. Geometric mechanics has provided many insights in enabling roboticists to understand robotic locomotion and manipulation in building novel systems. Some new tools used for combined use of dynamical systems theory & discrete mathematics. The techniques are used in networks of cooperating robots which can be viewed as dynamical systems on graphs. There is an increasing awareness of the hybrid nature of artificially systems and the fact that discrete & continuous dynamics which interact in complex ways. The mathematics of such hybrid systems is yet to see the light.

Robust models making use of uncertainty and to noise, and sensitivity analysis should be understood completely. There is no systematic approach to developing abstractions of dynamical systems and to decompose a dynamical system into a multi-scale hierarchy of subsystems. An emphasis on this set of issues will enable:-· Design and control of networks of coordinating, communicating machines for a wide range of application domains; · Multi-scale, distributed simulation of complex systems ranging from molecular and cellular networks to networks of interacting robots sweeping mine fields or performing assembly tasks; and ·new designs for agile, dexterous robots should be explored.

3. **Optimization algorithms.** Robotic system design and problems in robot task planning can be formulated as optimization problems. Some of them may be typically ``hard'' in terms of complexity and lack of readily recognizable or standard mathematical structures. Successful cases include graph-theoretic and calculus of variation based approaches in determining optimal paths. Also randomized algorithms for finding solutions in complex spaces, optimal feedback control policies for a range of robotic tasks, and saddle-point policies for solving differential games related to pursuit and evasion in respect of robotics should be investigated.

Sufficient effort must be made in formulating optimization problems of robotics. Firstly, the domains are usually non-convex and even non-smooth. Secondly, the solutions to optimization problems may require measure-theoretic tools from real analysis, and implementations of these solutions may provide constraints that are not easily incorporated. A good effort in this direction may yield new methods for developing - Optimal control and decision making policies, and · Algorithms for scheduling processes, allocating resources, and breaking of jobs will have wide range of applications.

4. Combinatorial Mathematics. Many robotic design and task planning problems involve aspects that are discrete. These problems are relating to motion-planning, involving analysis of a large continuous configuration space. The configuration space is approximated by a graph, and then discrete techniques are applied. The combinatorial complexity associated with these problem formulations seem, to be somewhat haphazard. Combinatorial mathematics is directly useful for Modular robots. These are robots built of many, identical modules. Such devices can adjust themselves to suit their environments. The configuration space of such robots is discrete. Discrete actuators are the smoothly actuated robotic arms which can be replaced by a cheaper network with devices that extend to two positions. It may be noted that arms are typically defined by fourteen

ISSN: 2278 – 7798

different parameters viz., Number of Axes, Degrees of Freedom, Working Envelope , Working Space, Kinematics, Payload , Speed , Acceleration, Accuracy , Repeatability, Motion Control. Also robot's eye is defined by the parameters of the camera. Robot position and configuration by a raster image which is a continuous image of the environment, faster and can be achieved by using a modest number of discrete sensors.

5**. Differential algebraic inequalities.** Discrete and continuum mechanics play an important role in the modeling of multi-body systems which are also important for robot manipulation. These systems are governed by differential algebraic equations (DAEs). In order to be able to model unilateral obstacle constraints, these DAEs have to be augmented by inequalities. The resulting differential algebraic inequalities are special cases of differential inclusions where in the computations and analysis require new mathematical theories and numerical methods. Qualitative behavior, which is useful for modeling robustness issues use simulation with uncertainties and a non--smooth analysis.

6. Statistical learning theory. Statistical approaches to robotics explicitly represent a robot's uncertainty in perception and action selection. They are highly robust to uncertainty and dynamic environments. As robots move away from factory floors into environments populated with people use of concept of uncertainty is a must. In recent times a class of hard robotics problems has been solved using statistical algorithms. A good research seems to be the need of the hour along multiple dimensions to develop more robust and more efficient algorithms. We also need research on basic representations of uncertainty in robotics domains, along with fast algorithms for reasoning under uncertainty. We need better procedures for integrating learning into probabilistic robotics. A good amount of theoretical work is needed to expand the mathematical frameworks under which today's algorithms operate. There is a need to develop programming tools that facilitate the development of statistical algorithms for mobile robots. In brief the algorithms may be based on discrete geometry, Topology, algebra, dynamical systems theory, graph theory, geometry, optimization theory and probability theory[1,7].

8. Motion Queries, Planning and Simulation: The modeling of motion for robots include proximity queries, motion and or manipulation planning, simulation of a dynamical system, motion generation and execution, acquisition of kinematic structures from motion sequences, motion capture and display of changing environments (visual, haptic and auditory rendering). A good number of problems are relating to algorithmic robots and much has been studied. However, there are still several open research issues that discrete geometry and topology which potentially advance the state of art.

V. RESEARCH TRENDS

One possible approach is to simplify the geometric representations of the models is that of reducing the intrinsic complexity of the problem. Can we design a new class of algorithms based on multi-resolution (or progressively simplified) representations of the models to accelerate the performance of proximity queries? We can use complex geometry that is non-convex and non-rigid. This line of research requires multi-resolution analysis and approximation theory, in addition to discrete geometry. Analysis and understanding of configuration space for motion planning requires topology [1, 19,20].

A very fast algorithm can exploit graphics hardware to compute a very interesting generalized **Voronoi diagram** of relatively complex environments of arbitrary geometry. The complex environment may consist of hundreds of thousands of primitives. We are interested in how we can use the Voronoi skeletons of the 3D workspace, the dimensions of the robot and the narrow passages, and the kinematics of the robot to design a complete and fast roadmap method for real-time planning in a dynamic or partially known environment. Some problems relating to planning for non- rigid robots, assembly planning of MEMS and nano-structures, planning and coordination of multiple autonomous agents, planning with constraints, very high degrees of freedom, uncertainty, etc are potential areas for research.

Dynamic simulation of rigid and non-rigid bodies is mentioned to be potential research area for modeling, analysis and understanding of physical interaction between the robot and the environments.

ISSN: 2278 – 7798

Modeling of soft and flexible bodies for the design and planning of medical robots, as well as surgical training using haptic interface is one area to be explored.

Mathematical tools required includes discrete geometry for proximity queries particularly dealing with non-convex and deformable bodies, numerical analysis, constrained non- linear optimization techniques, etc.

Sensor Placements: The problem of sensor placements requires configuration graphs, linear programming, and finding shortest paths using Dijkstra's algorithms. Some situations where in we can use shortest path yet not studied relatively easily deal with N-bit factor. But there are some enormously large graphs with some structures, where we do not explore the entire graphs. One possible approach is getting an embedded graph that can give a near-optimal solution. The characteristics in a graph structure differentiate one graph from others? If the graph is completely unknown, then the shortest path approach is the only one. Probabilistic techniques should also be considered.

Self-Configurable Robots & *Counting polyominoes*: Self-configurable robots can change shape based on sequences of simple translational and rotational motion. Self-configurable robots require planning of motion sequences to change from one configuration to another configuration. For lattice-based self-configurable robots, the configurations can be thought as finite connected sub-graphs with respect to a lattice graph. These sub-graphs are called polyominoes. We can study complexity of such a problem. That is how many possible ways can possibly have to reconfigure the robots, so that we can change its shapes between one configuration to another. While the known algorithms requires $O(n^3)$ time for counting polyominoes where n is the number of modules. But the foundations of graph theory and other discrete geometry tools are of immense use..

Planning algorithm for modular self-reconfigurable robots: We need to consider the size of obstacles and robots for using self-reconfigurable robots for planning. The current system is lattice based. Research is also required about simpler prototype for basic understanding, like ameba-like or hexagonal modules. We need to study shape representations in three dimensions for better, general understanding. A lower degree-of-freedom system to minimize the mechanical engineering complexity of projects is interesting, where we can wrap a rope for a stack of cubes and move the cubes by pulling the robe to derive an analytical solution for describing the perimeter of this rope which forms the convex hull of the given collection of cubes is interesting.

Manipulation in unknown environments: Some examples include rescue missions for plane crash, bomb squad mission, and working in hazardous environments. When sensors are given incomplete information, how do we plan manipulation? Sending coordinated groups of robots vehicles for hazardous missions is another problem. If some type of mathematical formalism be useful, then it must be consisting sensor design.

Motion Planning and Kinematics: Motion planning is exponentially hard, so there is the so-called probabilistic road map (PRM) that seems to work well practice in some scenarios, but not in some cases. So, we probably need to look at lower dimension problems to find a complete solution that is output sensitive. Projection down to lower dimensions recursively and topological analysis may be the keys. Also many possibilities may involve robot kinematics maps to some sort of graphs and the connectivity of the configuration space is encoded in some sort of graphs[12,13].

Topology Overview: Interaction with mathematicians may or may not yield the precise tools needed to solve a given robotics problem. However, it is almost certain that interaction will change the perspectives that robotics researchers bring to the table. Mathematics will affect the question as much as the answer. Each of the disciplines discussed in this section possesses one or more natural features of immediate relevance to solving real-world problems: Topology is implicitly robust and global; Dynamics links topology to the concepts of stability and attraction; Geometry is perfectly suited to answer questions about efficiency and coordinates. Given the nature of applications desired, it is important to emphasize the computational aspects of the methods used. In particular, ties with the inchoate discipline of computational topology should be established.

Specific mathematical ideas/tools: The following ideas and tools have potential to be useful in new ways to robotics research. Path/loop spaces --- and related homotopy-theoretic tools --- should be helpful in C-space analysis.

ISSN: 2278 – 7798

Morse theory in all its various permutations: · Morse and stratified Morse theory are already of use in C-space analysis to a limited extent. One goal is to increase the utility of these powerful subjects. Conley-Morse theory is proving to be increasingly useful in dynamics applications. Perhaps there are problems in robotics for which these techniques would be of help. Geometric group theory and non-positively curved space technology can generalize hyperbolic geometry to groups and metric spaces. Can these help in C-space problems and path-planning? These ideas (NPC, CAT(0), etc.) all are tied to problems of computability (e.g., solving the word problem in group theory, etc.) --- this may be of help in path planning problems. Automatic groups should be of use in manipulation and planning.

.

Combinatorial differential topology: There are precise notions of differential objects (vector fields, Morse functions, differential forms) for combinatorial cell-complexes. These are supposed to be well-suited to computation yet accurately capture the topology (homology/co-homology) of a complex. This may be one method for using smooth techniques while storing all data combinatorially.

Computational homology: Several groups are working on fast algorithms for computing homology of maps on simplicial complexes. These have the promise of working in relatively high dimensions, or, at least, in real-time for low dimensions. The tools should certainly be of use to vision problems (e.g., checking connectivity or changes in global structure, detecting occlusions, etc.).

Distributed robotics: Work on Distributed robotics started in the 1970s and algorithms are developed in an architecture dependent way on a task by task basis, but formal foundations will make algorithms more constructive. In the early 1980s the realization that uncertainty was the fundamental obstacle in robot control which leads to the invention of configuration space, and cause a qualitative leap in the advancement of robot science. As sensors and actuators are becoming smaller and cheaper, teams of robots working together are becoming more pervasive.

Cooperative robotics has the potential of expanding greatly the application domains, are not completely explored field. In addition to all the challenges of single robot systems, cooperative robotics has the added difficulties of cooperation and communication, and of combining discrete and continuous systems. This is an opportunity for the development of a mathematical basis for distributed robotics.

In computer science, one often learns a lot about the structure of an algorithmic problem, by analogous methodology which may be useful in robotics research also. The three most important challenges in distributed robotics, where mathematics is likely to make significant advances are: 1) Developing formal models that allow principled comparisons between distributed robot algorithms and give performance guarantees; 2) Developing dynamic control for non-smooth systems; and 3) Developing methods to characterize the power of modular robot systems and of matching structure to task.

Formal Models for Distributed Robotics: The generation of distributed robot algorithms with performance guarantees, seem to be a fruitful area. Structured environments, such as those found around industrial robots, contribute towards simplifying the robot's task because a great amount of information is encoded, into both the environment and the robot's control program. These encodings are supposed to be difficult to measure. A possible solution is to quantify the information encoded in the assumption that (say) the mechanics are quasi-static, or that the environment is not dynamic. One may ask how much ``information'' must the control system or planner compute? Successful algorithms exploit mechanical computation, in which the mechanics of the task circumscribes the possible outcomes of an action, which use physical laws. Planning or simulating these strategies may be computationally expensive. Since during execution we may witness very little "computation" in the sense of "algorithm," traditional techniques from computer science have been difficult to apply in obtaining meaningful upper and lower bounds on the true task complexity. It will be interesting to develop the concept of information invariants for characterizing sensors, and the complexity of robotics operations, similar to the Turing model developed for computation. In computational geometry it necessary to develop a measure for characterizing input sizes and upper and lower bounds for geometric algorithms. This measure seems somewhat relevant in embedded systems, which is a reflection of change in the scientific culture. In the off-line model, we might

ISSN: 2278 – 7798

assume that the robot, on booting, reads a geometric model of the world from a disk and proceeds to plan. The online agents are not assumed to have an a priori world model when the task begins. Instead, as time evolves, the task effectively forces the agent to move, sense, and (perhaps) build data structures to represent the world. From the online viewpoint, off-line questions such as “what is the complexity of plan construction for a known environment, given an a priori world model?” often appear secondary.

Modular robots: The goal of modular robotics is to create more versatile robots by using reconfiguration: hundreds of small modules will autonomously organize and reorganize (automatically or manually) as geometric structures to best fit the terrain on which the robot has to move, the shape the object the robot has to manipulate, or the sensing needs for the given task. The science-base of modular robotics is in an embryonic state, but there are lots of opportunities with huge potential impact for applications. Several mechanisms capable of reconfiguration have been proposed, and a few centralized planning algorithms: the start and goal configuration are specified and a global planner and controller synthesize motion sequences that move one module at a time. Some of the most interesting applications of this work will employ thousands of modules working together. Such systems constitute ultra-high degree of freedom systems. The off-line planning algorithms proposed above move one module at a time and may be too slow and impractical for controlling lattices made of thousands of modules. They do not take advantage of the natural distribution and redundancies in the system and presume too much global knowledge. Our goal for this part of the project is to develop distributed planning algorithms that are highly parallel, use local communication to neighbors, and minimize the required global information.

In a distributed approach modules make local decisions on-line about where to move (in parallel) in each round. Complications arise from the very high degree of freedom and under-constrained nature of these systems. In addition, there are constraints on the connectivity and on the static and dynamical stability of the structure. Moreover, since these robots are not connected in a fixed topology (but rather, are allowed to reconfigure in response to tasking demands) the control, coordination, and programming of such devices remains very challenging. Finally, the control issues required by a physical implementation include synchronization at many levels: the clock used by each system, making connections, and the actual motion, which is difficult. This leads to natural problems in dynamic control.

Another important aspect of modular reconfigurable robots is the characterization of exactly what such robots could do. The modules can aggregate with other identical modules and they can move relative to the world, but what exactly can they do, and how long does it take? We would like to characterize the class of static and dynamic objects that can be assembled from crystalline modules. For example, algebra could play an important role in characterizing the class of structures and motions that can be achieved with a given unit module. An interesting line of investigation is to treat each module that forms the basis of a reconfigurable robot.

Motion trajectories correspond to specific paths in this subgroup according to the specific types of actuation used in the module. We would like to characterize the nature of these subgroups, and develop a formalism that makes it easy to examine different modules with different types of actuation, and compositions with modules of different types.

Dynamics and control In mechanics, systems may be holonomic or non-holonomic. In general, a holonomic constraint is a wholly integrable sub-bundle E of the tangent bundle. The system outcome for a non-holonomic system is path dependent. Non-holonomic systems have been studied in robotics. Examples include: car-like robots, tractor-trailers, bicycles, roller-blades, airplanes, submarines, satellites, and spherical fingertips. In robotics, a non-holonomic system is usually defined by a series of non-integrable constraints of the form $G_i(p,v)=0$ on the tangent bundle. For example, whereas holonomic kinematics can be expressed in terms of algebraic equations which constrain the internal, rotational coordinates of a robot to the absolute position/orientation of the body of interest, non-holonomic kinematics are expressible with differential relationships only. This distinction has important implications for the implementation of a control system. It is well known that many under-actuated manipulation systems are naturally modeled as non-holonomic systems. An important

ISSN: 2278 – 7798

control problem is the analysis of isotropic non-holonomic manipulation system in which multiple robots manipulate objects by using non-prehensile grasps, or by using prehensile grasps enabled by tools such as ropes.

Non-smooth analysis: Non-smooth analysis has reached a certain level of maturity in the last 10 years but still remains relatively inaccessible to roboticists, in part due to the limited communication between the areas. Consider the problem of finding the zeros of functions which are PL smooth, or maximizing an objective function which is also non-smooth. These problems have been around for 40 years without much change. But in last 10 years, new assumptions and algorithms have become available. Many people who actually pursue this within the mathematics community are somewhat marginalized but the potential interaction with robotics has the potential to speed up the development here, with consequent advantages for both areas.

Example problems include: collision problems, contact problems, friction problems, grasping, manipulation, sliding, and jumping. In a different vein, challenge coming from other areas of robotics with requirements in same area of non-smooth analysis. Examples include hybrid systems, with event driven components, e.g., motion planning in changing environments, such as assembly.

Modeling of uncertainty in robotics, propagation of errors in map building for local robots in environments which are not completely known. If we use odometry in map-building, how do we update and correct for errors? How do errors accumulate to errors in linkages? This is relevant to assembly tasks. How does one acquire the bounds for errors, and figure out how to make corrections? Mathematicians in this area can well have their interests focused by the types of problems that occur in this way in robotics areas.

Probability theory and statistics are clearly the dominant tools in a number of areas: Images coming from a sensor, motion sensing and pattern recognition. Activities for the mathematical community arising from these issues include probability theory on groups, and solving minimax problems. For reconfigurable robots: just to change shape we get into combinatorial manifolds of extremely high dimension. Roboticists don't know anything about them, and mathematicians in fact do have knowledge in these areas, but not at the level of detail required for these applications. When dealing with sensors or actuators - when we make a movement many muscle groups are brought into play, some produce fine motion, some gross effects. This type of breaking down of tasks in terms of scale is fundamental to problem solving in the biological world. In engineering world, look at signal and extract information at these same levels of scale. There is a need for multi-scale development in robotics and engineering. Robotics could be a very useful source of problems. Currently, in fluid flow, turbulence is similar in terms of these effects. Could roboticists challenge mathematicians to focus on these problems?

Contact dynamics: There have been numerous developments in contact mechanics and dynamics in the past decade, and robotics has played a major role stimulating the development of these areas in mathematics and engineering. Starting with the work of Moreau and Monteiro-Marques in the 1980's and early 1990's, new and mathematically rigorous formulations of rigid-body dynamics with impact and friction have become available. This has been closely tied to new simulation techniques based on non-smooth analysis which have been developed by Moreau, Monteiro-Marques, Jean, Paoli, and Schatzman, Stewart and Trinkle, Anitescu and Potra, and others, on the mathematical side. There has also been intense interest in this area amongst the optimization community as well. The problems of dealing with impact mechanics for elastic bodies also has received a great deal of attention recently by Jarusek and Eck, Cocu, Raous, Martins, and others, although most of this work is being carried out in Europe.

However, there are a number of issues that have not been properly addressed. In rigid-body dynamics the impact "law" is an "input" into the simulation, rather than a consequence of it. Recent experiments and simulations (e.g., by Hurmuzlu and Stoianovici) indicate that the standard approaches to impact laws for rigid-body dynamics are inadequate, and elastic behavior must be incorporated. The mathematics of elastic bodies in impact is also very much under-developed, and even the existence of solutions and energy conservation issues are in doubt at present.

Two central issues remain in this area: · we don't yet have rigorous and useful results for elastic bodies in impact, with or without friction. - Full elastic-body models are computationally very expensive; we would like to develop low-order but

ISSN: 2278 – 7798

physically accurate models of impact. These issues may be resolved in the next decade, but it will require a substantial effort and some new breakthroughs in the area.

Interactions: We need to add interactions with modern topology and geometry. Here, as regards the first issue, mathematicians developed many successful techniques for doing the analogue of differential equations (flows) on piece-wise linear manifolds 35 years ago in these areas, when they were studying the global classification and structure of manifolds. In particular, a critical component of the results of that time involved a corresponding notion, that of the tangent micro-bundle.

More recently, there has been an example of the potential for this area in robotics. Even the simplest kinds of problems involving motion planning for complicated (multi-link arms or two three or four link arms working in concert in an assembly environment) could not be handled since the structure of the configuration spaces was not understood. But using techniques in areas coming from Morse theory, Morse flows and handle decomposition, real progress on some of these problems has become possible, and some complete programs for motion planning with some of these mechanisms are now possible. Mechanics and Design Robots are electromechanical devices. Issues in the design of robotic systems, the kinematics and dynamics of their operation, and the physics of their interactions with the surrounding environment have preoccupied robotics researchers and practitioners since the earliest days of robotics research. There is a substantial accumulated body of work in these areas, and many problems are now considered solved. However, there are still important issues that will require long term attention and non-trivial mathematical expertise. Some of these issues arise because of ongoing advances in technology. The mechanics and design session identified the following broad themes and individual problems during its discussions.

Robotic System Design and Design Methodology: Electromechanical design is a formidably complex problem as the dimensionality of the ``design space'' (the space of all possible choices for system parameters) is so huge 106 materials, 102 fabrication methods) and the functions that describe design performance are highly non-smooth and highly non-convex. The development of formal methods for design is an active research area in the mechanical design and circuit design communities. Formal and automated design methods have been very successful in VLSI system design because the problem is highly constrained. Formal methods in mechanical design have been less successful, possibly because the design possibilities are so vast. The possibility of bringing greater mathematical rigor and fresh mathematical ideas to this subject is promising.

It was generally felt that advances in design methodology might be made by constraining the focus to a more narrowly defined set of problems. In particular, structured and automated design methods for MEMS (Micro-Electro-Mechanical Systems) is attractive, as the design space of material (Si, GaAs, etc.) and fabrication techniques is smaller. MEMS is also likely to have a strong impact on robotics in the future in terms of cheap, distributed sensing and possibly actuation. Hence, advances in automated MEMS design capability will have a positive impact on robotics. The MEMS community is now paying some attention to this issue, but it the subject is still in its infancy. Robustness and uncertainty is a common theme that has arisen throughout all areas of this workshop. It is impossible to manufacture components precisely. For example, even in the highly mature disk drive industry, disk drive micro-actuators are fabricated with 15% variation in parameters. Design methodologies that better account for these variations are clearly needed. The discussion also highlighted the following important trends and their associated needs for new research.

It is now clear that some form of household robotics will become widespread in the future. A general issue is how to use robots to improve quality of life, particularly for disabled people. In the design context, the issue of how you design robots to work closely and safely with humans is a largely untouched subject. Future robotic systems will be distributed and comprised of many interacting components, or “agents.” Examples at the macro-scale include “Intelligent Transportation Systems” while meso-scale and micro-scale examples include distributed part manipulation. Design of large-scale systems to enable their coordination is a critical point. Distinctly new design methodologies are needed for this domain. Moreover, one cannot decouple mechanical design from control and sensing in these problems; it is truly an integration problem.

ISSN: 2278 – 7798

Because of advances in micro- and nano-scale fabrication, future robotic systems will be comprised of components having vastly different scales in terms of size, energy usage, and computation communication needs. Design methodologies for integration of micro and macro systems/components are desirable. Along these same lines, multi-scale modeling is an attractive objective. · Robots are widely used in the nuclear industry. As they are extended to other hazardous and extreme environments, such as search and rescue, new design paradigms are needed. Finally, the goal of developing systems with minimalist actuation by exploiting non-holonomic constraints to design under actuated robots seem to be a promising area.

Optimization and Simulation: System design is tightly linked to optimization. With recent advances, optimization problems can be solved for millions of parameters, even with constraints. Computations associated with design are an attractive opportunity. Also the associated optimization problems will only be solvable by numerical techniques.

The approximation methods makes research into investigation of set-based design, simulation, and optimization methods also will be an interesting area. In set-based simulation one simultaneously simulates a whole set of systems, and the output is not a trajectory, but will be a set of trajectories. Simulation of hybrid systems is a related and promising topic. Interval-based methods, interval arithmetic, and automatic differentiation will be useful tools.

Genetic algorithms and ``evolutionary'' computations and design techniques have been effectively used in large-scale system design. They produce a robust solution, but not necessarily an optimal one. A greater theoretical understanding of these algorithms is needed.

Self-Organizing and Reconfigurable Systems: George White team has developed impressive demonstrations of self-organizing mechano-chemical systems. In these demonstrations, simple shaped objects are selectively coated with hydrophilic/hydrophobic molecules. Through different mechanics, such as evaporation, vibration, etc., these systems self-assemble into complex shapes and networks. There may be several interesting & beautiful mathematical results in this area. Within the robotics community, there has been much recent work on modular and reconfigurable systems. The lack of tools for modeling and design of reconfigurable systems, as well as design of limited (degrees of freedom) DOF systems that are reconfigurable hamper real progress in this area.

Kinematic Synthesis: Kinematic synthesis is an enabling technology for robotic systems. There are obvious opportunities to apply methods from algebraic topology to the kinematic synthesis problem; particular in the case of closed-chain mechanisms.

Bio-mimetics: Borrowing concepts from biological systems as a basis for engineering system design has proven to be an appealing concept. However, ``biomimetics'' is still more of a philosophy than a rigorous discipline. Within the broad realm of biomimetic system design, there appear to be targets of opportunity for mathematical analysis that leads to useful systems. For example, fish locomotion is mathematically complex phenomena whose greater understanding could be translated into design of new systems. As molecular biology advances, future design paradigms must also consider biological material as a potential design substrate.

Contact Modeling: Contact modeling continues to be a vexing problem. New contact models and their experimental validation is an issue. Soft Tissue Modeling and New Modeling: It is now clear that robotics will have a significant impact on medicine. To support anticipated medical applications, modeling of soft tissues and organic structures is needed. Such models are needed for design of medical robotics and for simulating actions. Because of the highly anisotropic and large deformation characteristics of soft materials, their faithful simulation is difficult. Low order models are appealing, but possibly un-realistic. As robots increasingly interact with more diverse environments, we need to move beyond rigid body and mechanical models. We need a much richer set of models of how the world behaves (mechanics), more phenomena (deformations, soft tissues) and new domains.

Design Metrics: How do we compare systems that solve the same task but use different sensors related to vision. These cases are needed for design studies.

Uncertainty: Uncertainty is pervasive throughout robotics. A vast majority of existing motion planning and control algorithms in robotics does not

ISSN: 2278 – 7798

take uncertainty into account. A proper handling of uncertainty will almost certainly lead to the study of robust systems. A better understanding on how to perceive and act in the physical world is the need. The importance of uncertainty will increase as robots move away from factory floors into increasingly unstructured environments occur in domestic circles.

Robotic surgical systems face an enormous amount of uncertainty, which is currently only marginally considered during motion planning, manufacturing, drilling, tunneling and robotic exploration as it contains uncertainty. Hence it would benefit from better and mathematical and computational tools are necessary to make decisions under uncertainty.

The range of complimentary frameworks exists for representing uncertainty: Probabilistic methods (which include parametric and non-parametric representations), binary representations, Dempster-Shafer logic, fuzzy set theory, and others. The choice of the representation influences the difficulty of crafting models and the computational efficiency of using these models. Additionally, a range of different problems can be attacked under uncertainty, such as: prediction vs. planning vs. control; worst case vs. average case; and correctness vs. optimality.

A list of questions to be studied is --

How can uncertain information be propagated through process models, and what type of bounds can be obtained?

How can we devise systems that can reason about when and what to sense?

How can we develop contingency plans that interleave sensing and control?

How can we reduce the complexity of probabilistic propagation and planning?

How can we approximate uncertainty and devise bounds for those approximations?

What problems can be solved in closed form, and which solutions can be computed efficiently?

How can we best represent geometric uncertainty, or shape uncertainty?

How can we devise planners that employ feedback mechanisms for reducing uncertainty?

How can we, on solid mathematical grounds, ascertain which uncertainties can be ignored?

The fields of statistics, operations research and computer science has long addressed some of those issues, though typically deprived of a robotic context and in relatively small worlds. Current theory often focuses on discrete problems, whereas robotics spaces are typically continuous.

Examples include “Optimal Experimental Design,” “Partially Observable Markov Decision Processes,” “Sensitivity Analysis,” and “Monte Carlo Methods.” We urgently need research to develop these and similar fields in the context of robotics. A work to make these basic methods amenable to complex robotics problems and enable the deployment of robotic systems in uncertain, real domains. Computer Vision Computer vision has witnessed rapid progress in the past fifteen years, with great success in classical application fields like industrial automation, and, more recently, emerging areas such as medical robotics and the entertainment industry. State-of-the-art industrial vision systems rely on sophisticated mathematical tools from geometry and statistics to model the practical problems they address. In addition, fundamental aspects of computer vision, such as the analysis of image sequences of rigid point sets observed by a roving camera, or the characterization of the appearance of smooth surface outlines, are now fairly well understood.

The ever-increasing computer power, tackling harder and exciting problems such as building general-purpose visual inference machines has become a realistic endeavor. Yet, it is unclear how current approaches can be generalized to solve such difficult problems: indeed, computer vision has sometimes been criticized for its lack of formal foundation and empirical justification, and, in contrast with mature disciplines, like control theory and information theory, it certainly lacks a mathematical framework.

Such a framework should include a common language for describing vision problems, mathematical techniques for modeling them, and algorithms that can be applied to solve them. For example, the visual recognition of learned object models is a key problem, maybe *the* key problem, in computer vision. Solving it will require radical advances in object representation, image segmentation, and in our understanding of the recognition process.

Probability theory and statistical inference form a promising foundation on which to build a mathematical framework for object recognition, but

ISSN: 2278 – 7798

major conceptual and technical difficulties remain: for example, how can we define object models that are easy to learn and effectively support inference from pictures? How can we construct algorithms that will perform Bayesian inference in real time? How should we handle large numbers of objects and object classes? There has recently been encouraging progress on these issues following the introduction of techniques from statistics, mathematics, and information theory in our field. Conversely, the demands of computer vision throw tough challenges and interesting problems at these disciplines. Indeed, as argued by Mumford (2000), the type of statistical inference required for computer vision may become a central theme for mathematics in the twenty first century. Likewise, we believe that a strong synergy between areas of mathematics such as geometry and topology, recent advances in computer technology, and effective engineering practice, will result in revolutionary advances in the theory and applications of computer vision to areas as diverse as visual robot navigation, medical robotics, and automated three-dimensional model acquisition for the entertainment industry,.

VI. CURRENT RESEARCH FOCUS:

The study of basic engineering problems is central to our work, but equally important is the integration of innovation and discoveries into real-world systems and applications. The current trends in Robotics Research is on the subtopics Human-Robot Interaction & Interfaces, Computer Vision, Embedded Systems for Robotics, Motion Planning & Control, Machine Learning for Robotics, Medical and Assistive Robotics , Human Augmentation, Novel Sensors/Actuators And Robot Design, Soft Robots, Multi-Robot Systems.

Robotics deal with mechanics, control, perception, AI & cognition (human behavior and emotion, hardware and software brains), interaction **a**nd systems, effectors and mobility (autonomous flying vehicle project, fish robot, muscles, robocop), sensor detection (robotic vision, sensor based motion planning), control systems (hierarchical behavior control, nano-technology and medical applications, intelligent systems for communication networks, active vibration control, hyper-redundant robotics systems[14,15,16,17,18]. These are only few areas suggestive to carryout theoretical as well as practical designs. We need to educate students, guide and involve in team effort to pursue the investigations, so that we proclaim that India has the capability to design and implement.

Hence the list of the actual software used is of greatly large in number. Many organizations like Math-works, wolfram.

VI. REFERENCES:

1. Richard M. Murray, Zexiang Li, S. Shankar Sastry; A Mathematical Introduction to Robotic Manipulation, 1994, CRC Press; http://www.cds.caltech.edu/∼murray/mlswiki
2. Emilio Frazzoli, Tomas Lozano-Perez, Nicholas Roy, Daniela Rus; (Ed) Algorithmic Foundations of Robotics X: Proceedings of the Tenth Workshop on ..Algorithmic foundations of robotics , springer, 2013.
3. David Kramer, NASA sees a hat trick in its robotics program; August. 2015.
4. Hugh Durrant-Whyte and Tim Bailey; Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms. www.cs.berkeley.edu/.../Durrant-Whyte_Bailey_SLAM-tutorial-I.pdf.
5. J. Andrew Bagnell, Reid Simmons, Rachel Burcin; Summer Scholars Working Papers Journal is an annual publication of the Robotics Institute’s Summer Scholars Program at Carnegie Mellon University. Carnegie Mellon University 2013.
6. Sebastian Thrun, Wolfram Burgard, Dieter Fox; A Real-Time Algorithm for Mobile Robot Mapping With Applications to Multi-Robot and 3D Mapping; EEE International Conference on Robotics and Automation, San Francisco, April 2000.
7. S. Thrun, ,M.Beetz, , M. Bennewitz, W.Burgard, ,A.B.Cremers, , F. Dellaert, D. Fox, D.H ̈ahnel, C. Rosenberg, ,N.Roy, ,J.Schulte, ,D.Schulz, Probabilistic Algorithms and the Interactive Museum Tour-Guide Robot Minerva; Journal of Robotics Research ***robots*.*stanford***.*edu/ papers/thrun.ijrr-minerva.pd.*

ISSN: 2278 – 7798

8. History of Robotics: Timeline; Robot-Shop Distribution Inc., 2008; www.robotshop.ca; usa.
9. A Brief History of Robotics – Mega-Giant Robotics; robotics.megagiant.com/history.html
10. www.imperial.ac.uk/robotics/
11. mime.oregonstate.edu/robotics-area-research-excellence,
12. Ideals, Varieties, and Algorithms: An Introduction to Computational ... By David A Cox, John Little, Donal O'Shea.
13. S. M. LaValle ; Robot Motion Planning: A Game-Theoretic Foundation; Algorithmica
14. Glossary of robotics, From Wikipedia, the free encyclopedia
15. 99 Superb Sites on Mechatronics & Robotics Engineering. www.electricalengineeringschools. **org/mechatronics/**
16. NASA Robotics – Links: Resources – Robotics Alliance Project, robotics.nasa.gov /links /resources.php.http://www.autopenhosting.org/robots /research.html;
17. Resources for Robotics Research - The Robot Group; therobotgroup.org/resources-for-robotics-research/
18. Research Websites - Free Collaborate Research Tool, Adwww.piirus.com/Research.
19. World's Largest Robotics Sites
20. The Interplay between Mathematics and Robotics; summary of workshop; National Science Foundation Arlington, Virginia May 15, 16, and 17, 2000.
21. White sides Group Research; soft robotics.2011.
22. Variational Inequalities and Frictional Contact Problems Volume 31 of the series Advances in Mechanics and Mathematics pp 1-6, Date: 23 August 2014.
23. Robotics / types of Robots – Wikibooks.

ISSN: 2278 – 7798