Chapter 2

Literature Review

This chapter provides further justification for the significance of the research objectives identified in Chapter-1. A critical review of the literature, relevant to the work reported in this thesis is described in this chapter. This chapter also presents the background knowledge that facilitates discussion in subsequent chapters. Section 2.1.1 presents a brief description of the important terms related to time synchronization. Section 2.1.2 presents a brief review of the different techniques and algorithms available for time synchronization in static and mobile wireless sensor networks. The suitability of these protocols for swarm robotic systems (SRS) or low-cost multi-robot systems (MRS) is analyzed in Section 2.1.2. The different technologies and techniques available for localization in indoor environments and their limitations are presented in Section 2.2. Section 2.3 describes the Multi-Robot Task Allocation (MRTA) problem and the available solutions for the same. The major research gaps related to time synchronization, localization and task allocation in SRS and MRS which are identified, are summarised in Section 2.4 of this chapter.

2.1 Time Synchronization

The mechanism to provide a common notion of time across the nodes in a network is referred to as time synchronization [32]. Time synchronization is a critical system maintenance task in any distributed network, as information obtained over a distributed network has significance only when associated with a timestamp and a location stamp. Maintaining a common time scale across nodes in a distributed network is also critical for a variety of other reasons such as identification of causal relationships among various events in the physical world, elimination of redundant data in the network, coordination of network functionalities, etc. Time synchronization is often a prerequisite for other layers of network protocol stack such as routing and Medium Access Control (MAC). A multi-robot system or a swarm robotic system should be time synchronized in order to be utilized for practical applications. In this thesis, the robot in SRS or MRS is referred to as a node and the terms 'node' and 'robot' are interchangeably used. Time synchronization can be broadly classified into two types- External and Internal synchronization [33]. External synchronization is possible when access to a reference source of time (e.g. Coordinated Universal Time (UTC)) is available in the network. In external synchronization, the nodes in the network seek to adjust their local clock (time) to an external reference source of time. In internal synchronization, a global time reference is not accessible to the system and the goal of time synchronization is to minimize the difference between the readings of local clocks of all nodes in the network. For networks such as SRS or MRS, access to the standard source of time may not be always available and the nodes are only required to maintain internal synchronization [32]. To achieve time synchronization, it is required that nodes in the network communicate with each other through wired or wireless communication links. In this thesis, the problem of time synchronization for SRS and MRS, which makes use of a wireless network for communication among its members is addressed.

Time synchronization in static wireless sensor networks is a widely investigated topic in sensornet community and several research works are reported for time synchronization in wireless sensor networks. Time synchronization for wireless mobile networks is a relatively new area of research. Time synchronization error can be a within a few hundreds of milliseconds for most of the static WSN applications, whereas swarm robotic systems require higher synchronization accuracy (lesser error bound, in the order of few hundreds of microseconds) because mobile robots (nodes) in addition to monitoring or controlling the environment in which they are deployed, may utilize timing information for their localization, co-operative path planning, navigation, aggregation, dispersion etc. Although several works are reported in the field of routing and medium access control for mobile wireless sensor networks, or for navigation in a collaborative swarm, prior time synchronization is stated as one of the assumptions when time dependency is involved [34–36] and thus most of the research in mobile wireless sensor networks and swarm robotics is still confined to simulation-based study. With the constraints such as limited communication bandwidth, limited battery power, limited processing power, dynamic topology, time synchronization for a dynamic network such as a swarm robotic system or a cooperative multi-robot system is a challenging problem to solve. A description on the basic terminologies relevant to time synchronization is presented in Section 2.1.1. A brief review of relevant synchronization techniques is presented in Section 2.1.2.

2.1.1 Basic Terminologies

A node in a distributed network is usually clocked using a quartz crystal, excited by a clock driver embedded into the node hardware. The notion of time (logical clock) in a microcontroller-based node can be maintained using a Real Time Clock (RTC) implemented in software or as a dedicated hardware/microcontroller peripheral. RTC can be maintained in software using a timer peripheral that generates an interrupt at specified intervals of time. The number of time intervals is counted and then converted to time. The goal of time synchronization is to minimize the difference between the readings of logical clocks of all nodes in the network. In practice, RTC accuracy depends on the accuracy of the quartz crystal oscillator and the granularity of the RTC determines the granularity of the time synchronization. To elaborate the concept of time synchronization we define the following terms referred to in this chapter.

Instantaneous Offset $\theta(t)$: The difference between the time reported by two nodes at a given instance is referred to as time offset or instantaneous offset [37]. The nodes in a network are to be synchronized such that instantaneous time offset among nodes is maintained within the limits as demanded by the application. If the time reported by the reference node and a given node in the network is $R(t)$ and $N(t)$ respectively, then their instantaneous offset ($\theta_R^N(t)$) is given as

$$\theta_R^N(t) = R(t) - N(t) \tag{2.1}$$

The instantaneous offset includes two components -

- Initial offset, (θ_0): When nodes are physically distributed it is impossible to start clocks of all nodes in the network simultaneously, leading to initial offset among nodes. Even after the compensation of initial offset among nodes, their instantaneous offset will increase with time due to frequency offset or skew.

- Frequency offset or Skew(α) : The clocks tend to run at frequencies slightly different from their nominal frequencies due to the quartz crystal properties (mainly depends on the 'cut' of the crystal). Skew is the difference in the frequencies of two clocks. Skew between two nodes N_1 and N_2 can be represented as

$$\alpha' = \alpha_{N_1}^{N_2}(t) = \frac{d\theta_{N_1}(t)}{dt} - \frac{d\theta_{N_2}(t)}{dt} \tag{2.2}$$

where,

$$\frac{d\theta_N(t)}{dt} = \frac{\theta_N(t+\tau) - \theta_N(t)}{d\tau} \tag{2.3}$$

The term 'skew' is generally used to express the difference in frequencies between an ideal clock and a non-ideal clock. If node N_1 has an ideal clock then $\frac{d\theta_{N_1}(t)}{dt} = 1$, and skew can be expressed as

$$\alpha_{N_1}^{N_2}(t) = 1 - \frac{d\theta_{N_2}(t)}{dt} \tag{2.4}$$

In practice, $\frac{d\theta(t)}{dt}$ of clocks deviate from 'unity'. In low-cost networks, reference node clock is also non-ideal and hence we use the term 'relative skew' to indicate the difference in clock frequencies between two non-ideal clocks in the network. Relative skew is denoted as α' in this thesis.

It is important to note that the skew/relative skew may change over a period of time due to changes in environmental conditions, vibrations on the crystal, ageing, power supply variations, etc. The sources that lead to variation in skew is extensively studied by several researchers and variation in temperature is identified as the most predominant factor leading to variation in skew [38, 39]. Nodes are typically clocked using crystals which exhibit a skew of 1- 50 parts per million (ppm) for a temperature range -10°C to +70°C (40ppm implies a variation of 40μs per second) [40]. This necessitates frequent time synchronization to keep the instantaneous offset among nodes within tolerable limits as demanded by the application. A discrete time clock model is desirable since the synchronization is typically achieved by time-stamped message exchanges. If represented as a discrete time model, the instantaneous offset measured by a node with respect to another node at n^{th} instant can be described as

$$\theta(n) = \sum_{i=1}^{n} \alpha'(i)\tau[i] + \theta_0 + w(n) \tag{2.5}$$

where θ_0, α', w are the initial offset, relative skew, and the measurement noise respectively [41]. Generally, synchronization is repeated at periodic intervals of time called 'resynchronization interval'. 'Resynchronization interval', T_{resync} can also be defined as the periodic interval at which the nodes communicate to obtain information about the reference node time. Presence of wireless communication channel between the nodes introduce errors in timestamping of nodes and consequent errors in time synchronization as is discussed in Section 3.4. If the timestamps of the spatially distributed nodes used to measure their initial offset or skew are not recorded correctly, then irrespective of the synchronization algorithm used at the application layer, the time synchronization process will not be accurate. To minimize the measurement errors, careful understanding of the node hardware and possible sources of errors in RTC timestamping is inevitable.

2.1.2 Related Works in Literature

This section presents a review of the widely accepted protocols designed for time synchronization of nodes in other allied fields of research and evaluate their suitability for swarm robotic systems. Time synchronization for mobile networks in GPS-denied indoor and outdoor environments is still a nascent area of research [42]. Vehicular ad-hoc networks (VANETs) utilize global positioning systems (GPS) or other infrastructure based services. [43]. VANETs do not have constraints in terms of power, computational power or memory as in MWSNs or low-cost robotic networks like SRS. Hence, an accuracy in the order of 'microseconds' in time synchronization is easily achievable in VANETs. Although several time synchronization protocols have been suggested for static wireless sensor networks, claiming accuracy in the order of few microseconds, most of these protocols incur unreasonable communication overheads, convergence time or computational complexity which make them unsuitable for practical applications. Improvement in resynchronization interval, obtaining deterministic synchronization accuracy over multi-hop networks, suitability for dynamic environments are still open problems of research in static WSNs. In this section, we review some of the widely accepted time synchronization protocols for WSN and evaluate their suitability for a scalable and dynamic network such as a swarm robotic system. The evaluation is based on the following metrics- scalability, resynchronization interval, adaptability to dynamic environmental conditions, robustness to loss of connectivity, changes in network topology, communication and computational complexity.

Receiver Broadcast Synchronization (RBS) [44], Timing-Sync Protocol for Sensor Networks (TPSN) [45] and Lightweight Tree-based Synchronization (LTS) [46] protocols utilize frequent, hierarchy based, pairwise message exchanges to achieve time offset compensation and do not implement explicit skew compensation mechanism. Offset compensation among a pair of nodes will take few milliseconds, as the process is severely restricted by the data rate of these networks ($\approx 250-500kbps$). By this time, the reference node which exhibits a skew of few ppm would have deviated from its ideal frequency. When this reference node is used for offset compensation of other nodes, it propagates cumulative "inter-sync" error resulting in large variance among clocks of nodes even at the same hop distance. Thus, the protocols which utilize pairwise two-way message exchanges, cannot be adopted for scalable and multi-hop networks such as a swarm of robots. RBS, a receiver-receiver protocol, requires communication among multiple receivers after the broadcast packet transmission from the reference node for synchronization. For mobile networks, the presence of receiver nodes in the vicinity cannot be guaranteed without imposing restrictions on the mobility of the nodes. Mobile nodes are prone to frequent hierarchy changes, hence protocols such as RBS, TPSN or LTS, which impose restrictions on the network deployment will introduce frequent level discovery overhead if used for synchronization.

Models that are approximate to receiver - receiver synchronization are defined in [47]. Maximum likelihood estimators (MLE) are derived from these models and the corresponding Cramer-Rao lower bounds are derived for time synchronization. These estimators are computationally complex requiring multiplication operations which cause frequent memory overflow and consequent synchronization errors if implemented on low-cost microcontrollers without a floating point unit (FPU). This technique also requires the nodes to be neighbour-aware, i.e., every node is required to maintain a list of neighbours that it can communicate with and thus neighbour discovery is a prerequisite for the implementation of this technique. The technique is memory intensive and with the requirement of neighbour discovery, the synchronization will incur large communication overheads and is thus not suitable for dynamic networks such as a swarm of robots.

Flooding Time Synchronization Protocol (FTSP) [48], PulseSync [33], Adaptive Linear Prediction Synchronization (ALPS) [49] utilize timestamps recorded at the sender and receiver nodes through one-way broadcast synchronization messages for reference clock prediction. LR and LP techniques are based on the observation that the instantaneous offset among two clocks changes linearly under stable environmental conditions. FTSP and PulseSync perform a linear regression (LR) on pairwise timestamps from the nodes for predicting sender's clock whereas ALPS employs linear programming (LP) technique on the timestamps. The synchronized nodes then broadcast synchronization messages to other nodes in the network. Nodes which are outside the broadcast radius of the reference node utilize the timing information from the synchronized nodes that are located closer to the reference node. ALPS suggests that the offset between any two nodes at a time, 't + dt' can be predicted using linear programming (LP) technique as a linear combination of their previous offset's measured at an interval of 'dt'. LR and LP techniques require pairwise timestamps from the reference and the given node at resynchronization intervals to predict the reference node clock or their instantaneous offset respectively. If the resynchronization interval (T_{resync}) is 'T' seconds, then the timestamps are to be recorded at an interval of 'T' seconds and require at least 1 message transfer/hop at resynchronization intervals for FTSP, PulseSync and ALPS. FTSP recommends a resynchronization period of 30s and reports a maximum absolute synchronization error of 6.48ms for a two-node scenario [48]. PulseSync reports an average synchronization error of 2.06ms/hop for a resynchronization interval of 10s [33]. ALPS protocol suggests a resynchronization interval of 2s and reports an average error of 19.33 ms/hop [49]. Although the results provided above are reported based on the experiments on different hardware, it can be noted that the three techniques recommend a resynchronization interval in the order of few seconds.

Kalman filter (KF) is widely recognized as a suitable choice for real-time estimation problems. Researchers have proposed Interacting Multi-model (IMM) protocol based on Kalman filtering which predicts skew by combining results from two clock models i.e. a constant skew model (for stable environmental conditions) and a constant velocity model (for dynamic environmental conditions) [50]. Additional information aided multi-model Kalman filter (AMKF) uses temperature information for skew compensation and reports performance improvement over IMM. AMKF reports a maximum synchronization error of 2ms among two nodes for a resynchronization period of 1500s. LR, LP and KF techniques require frequent timestamping at periodic intervals of few seconds, restricting the availability of communication bandwidth and hardware resources of a mobile node for other critical functions. Timestamping at uniform periodic intervals over a wireless medium is difficult to achieve in practice. The accuracy of LR, LP and KF based techniques will be adversely affected by non-uniform intervals of timestamping and conditions of communication failures or loss of synchronization packets. LR, LP and KF techniques are proven techniques for offline synchronization but the accuracy of the same, when performed on nodes (even advanced microcontrollers have at the best, only a FPU with single precision floating point data type) needs to be evaluated. Since LR, LP and KF time synchronization is implemented hop wise, the loss of accuracy in reference clock prediction due to the data type restrictions and cumulative time synchronization error over hops will necessitate frequent resynchronization of the whole network. Consensus based approaches for synchronization are fully distributed in nature and recently several time synchronization protocols have been proposed based on consensus theory. Average Time Sync (ATS) and Maximum Time Synchronization (MTS) protocols are based on average and maximum consensus respectively [51, 52]. In ATS and MTS, the nodes thus utilize their neighbour's time information to achieve global synchronization. Incorporating the idea of clustering and consistency theory, Clustered Consensus Time Synchronization (CCTS) protocol was proposed [53]. Cluster based Maximum consensus Time Synchronization (CMTS) protocol proposed in 2017 is based on the maximum consistency theory [54]. CMTS which is an extension of MTS technique is superior to other consensus based techniques in terms of convergence rate and communication overhead. CMTS utilize two-way message exchanges and hence, the time taken for intra-cluster synchronization is dependent on the number of nodes in each cluster. In addition to this, in CMTS, the convergence time of entire network depends upon the relative position of the cluster with maximum clock and other clusters in the network. Moreover, implementation of techniques to detect the cluster head changes, addition and removal of cluster members is required if CMTS protocol has to be utilized for mobile networks.

A comparison of the popular time synchronization techniques discussed in this section is presented in Table 2.1. Major requirements of time synchronization protocol for scalable and dynamic mobile networks like multi-robot or swarm robotic network can be identified as follows:

- **Accuracy:-** Time synchronization error can be a within a few hundreds of milliseconds for most of the static networks, whereas low-cost MRS and SRS applications will require higher synchronization accuracy (lesser error bound in the order of few hundreds of microseconds) utilize timing information for their localization, co-operative path planning, navigation, aggregation, dispersion etc.

- **Improved Resynchronization Interval:** - Most of the protocols available in the literature suggest a resynchronization interval in the order of few seconds as mentioned in Table 2.1. The wireless channel will be utilized by the robot for several other functionalities like

Table 2.1. Comparison of time synchronization protocols

Protocol	Resync interval (s)	Average Accuracy (μs)	No of Messages per hop/cluster with 'n' nodes	No of iterations for Convergence	Synchronization Technique	Cumulative error over multi-hop
RBS[44]	-	29.1	$n(n-1)/2$	- -	Pairwise 2-way messaging	✓
TPSN[45]	-	16.9	$2(n-1)$	- -	Pairwise 2-way messaging	✓
FTSP [48]	30	1.48	n	-	LR	✓
PulseSync[33]	10	2.06	n	-	LR	✓
ALPS[49]	2	19.33	n	-	LP	✓
AMKF[50]	1000	1000	n	-	KF	✓
ATS [51]	30	600	n	120	Consensus	✓
MTS [52]	1	100	n	212	Consensus	✓
CCTS [55]	1	30.2	$n+3$	31	Consensus	✓
CMTS [54]	1	≈ 5	$n+2$	126	Consensus	✓

localization, cooperative path planning, navigation, aggregation, dispersion, task allocation, task monitoring, etc. If the resynchronization interval is in the order of few seconds, then the robots will not be able to use the channel effectively for other functionalities. An improvement in the resynchronization interval is essential.

- **Robustness Under Dynamic Environmental Conditions:**- The synchronization scheme should be adaptive to the changes in environment, predominantly the temperature and should not incur frequent resynchronization of the whole network with varying environmental conditions.

- **Bounded and Deterministic Synchronization Errors:** - Most of the protocols available in literature exhibit linear or quadratic increase in synchronization error as the size of the network increases. It is essential that the synchronization error of all nodes within the network is deterministic and bounded. Bounded synchronization error will also facilitate the development of other layers of protocol stack such as MAC and Routing, localization and task allocation of robots.

- **Scalability:**- The accuracy of the synchronization should be ideally independent of the number of nodes in the network. Addition or deletion of few nodes should not necessitate resynchronization of network or drastically affect the accuracy of networkwide time synchronization. From Table 2.1 it can be observed that as the number of nodes in the system increases, the number of messages required for synchronization also increases, which is not a desirable characteristic in a scalable network.

- **Robustness to Topological Changes:**- Mobile networks are prone to topological changes. To maximize the utilization of robots, they should be allowed to move seamlessly across clusters without necessitating resynchronization or introducing overheads in synchronization.

From the literature survey, it can be concluded that new techniques are to be evolved for time synchronization of SRS and MRS to meet the requirements as mentioned above.

2.2 Localization in GPS-denied Indoor Environments

With the advancements in the field of the internet of things and ubiquitous computing, context based applications requiring location-based services are gaining more and more importance. Undoubtedly indoor localization is of interest in several application fields. Robots in MRS or SRS should have the ability to determine and report their positions if they are to be employed for any real-world application. More flexibility in their movement and more cooperative behaviours leading to increased productivity can be achieved if the robots can determine their positions in the area of deployment[19]. In the context of localization, the nodes which have known locations are generally referred to as reference or beacon nodes and the nodes which are to be localized are called target nodes. Localization is generally carried out in a robotic network or a mobile wireless sensor network in two phases- 1. Beacon based localization 2. Self-localization. In beacon based localization, the location of a node/robot is performed with the help of reference or beacon nodes/robots deployed in the field of deployment. Self-localization of a robot can be described as the process through which a robot determines its own position using the information available with the robot or the information collected from its onboard sensors, without using any information from reference nodes. However, self-localization techniques are prone to cumulative errors, necessitating beacon based localization at regular intervals of time, thereby enabling the robots to correct their position. Global positioning systems (GPS) provides the location information of an object with GPS receiver, in outdoors with an accuracy of 2-10 meters [32], provided the GPS receiver is in Line-of-Sight (LOS) of four or more GPS satellites. GPS does not work indoors and indoor localization can be supported by ad-hoc solutions, among which the most promising ones are based on wireless sensor networks. Due to the complex nature of indoor environments, indoor localization is always associated with a set of challenges such as high Non-line-of-sight (NLOS) due to the presence of obstacles like walls, doors, equipment, movement of human beings, etc and the associated scattering and reflection of beacon signals leading to multipath effects [56]. The development of technologies for indoor localization systems has improved in the last decade. Therefore, both the research and commercial products in this area are new, and researchers in academia and industry are actively involved in the research and development of these systems.

This section provides a comprehensive review of the technologies and techniques utilized in developing Indoor Localization Systems (ILS).

2.2.1 Localization Approaches

Based on how well, an object should be localized, the localization approaches are broadly classified into following categories: physical Vs symbolic localization, absolute Vs relative localization and fine-grained Vs coarse-grained localization [29].

1. **Physical Vs Symbolic Localization:** In physical localization, the exact physical location of the object is determined in reference to a coordinate system. Physical location is expressed in the form of coordinates, which identify the location on a 2D or 3D map. Symbolic localization provides only location information that refers to abstract predefined notions of place. e.g. in the laboratory, next to a class, etc. [57].

2. **Coarse-grained Vs Fine-grained Localization:** In fine-grained localization the nodes in the network can estimate their distance or angle (orientation) estimates to their neighbors with known positions, and thus infer their position. In coarse-grained localization, only proximity (connectivity) information is available. Fine-grained and coarse-grained localization are also referred to as range-based and range-free respectively.

3. **Absolute Vs Relative Localization:** In absolute localization the location of the nodes is computed in reference to coordinate system that is external to the network. In relative

localization, all nodes are localized in reference to a node within the system. For three-dimensional relative localization, the target node is required to identify its relative position co-ordinates (x, y, z) with respect to a reference node in the network with known location (x_r, y_r, z_r). Similarly, for a two-dimensional relative localization, the target node is required to identify its relative position co-ordinates (x, y) with respect to a reference node in the network (x_r, y_r) [58]. Relative localization is the preferred localization scheme in SRS and MRS.

In this thesis, relative localization or the process of determination of the position of an object in two-dimensional (x, y) or three-dimensional (x, y, z) space with respect to a reference point (x_r, y_r, z_r) in the field of deployment is referred to as 'Localization'.

2.2.2 Performance Metrics

The major performance metrics asociated with indoor localization systems are summarized in this section.

- **Accuracy:** Accuracy is the measure of closeness of measured position to the true position. Accuracy may be expressed in units such as millimeters, centimeters or meters. For example, GPS typically provides an accuracy of 15 meters.
- **Precision:** Precision is the measure of how consistent, the results are when measurements are repeated. In continuation of the example provide for accuracy, if the measured position is less than 15 meters for 95% of measurements, the precision is 95%.
- **Responsiveness:** Responsiveness is a measure of how quickly the estimation of location can be obtained. The responsiveness is generally measured in terms of milliseconds or seconds.
- **Adaptiveness:** Environmental changes may affect the localization accuracy and precision. The ability of the localization system to cope with these changes is referred as adaptiveness. An adaptive system ensures easier system deployment as it requires lesser calibration.
- **Offline Vs Online Computing:** Some ILS systems require offline computation to obtain the position. Some ILS also require frequent labor-intensive calibration. These systems can

be only utilized for known and accessible environments which requires only non-real time positioning.

- **Cost and Complexity:** The cost of positioning system vary based on the cost of technologies & techniques used in the localization system, communication bandwidth utilized by the technique, energy consumption etc. The complexity of algorithms required for positioning and the hardware required for implementation of the necessary signal processing techniques and algorithms are also important factors which needs to be analyzed along with the performance of the ILS system.

2.2.3 Techniques for Indoor Localization

Location detection techniques can be broadly classified into three categories: proximity detection, Radio Frequency (RF) scene analysis and triangulation as shown in Figure 2.1 [59].

- **Proximity detection:** Proximity detection or connectivity based positioning is one of the simplest positioning methods to implement. A set of reference nodes or beacon nodes are placed in the environment. Either the reference node will periodically transmit beacon messages or the target node when it needs to be localized, will transmit the beacon. Reference nodes include their position information and their unique identification in the beacon message. The target nodes will then estimate the closest node to itself thus achieving a coarse-grained localization. If target node emits a beacon, then the reference node which receives the beacon

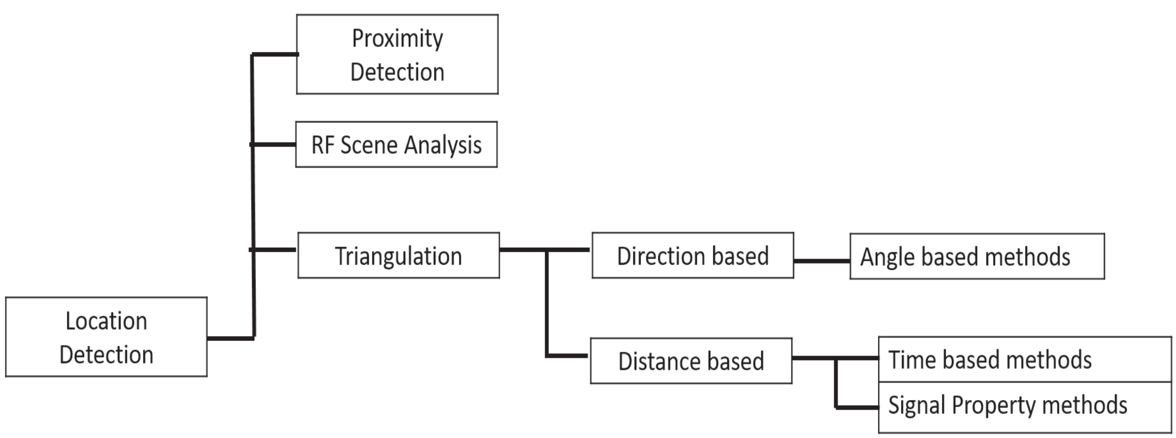

Figure 2.1. Techniques for indoor localization

determine the position of the target node based on its own position. When more than one reference node detects the mobile target, the system determines the position of the beacon based on the strongest signal is received. The accuracy of this technique is based on the density of reference node deployment and the communication range of reference and target nodes.

- **Scene Analysis/Fingerprinting:** This localization method is implemented in two stages: offline stage and online stage. During the offline stage, a site survey of the environment is performed and unique features of the scene (or fingerprints) are collected. During the online stage, the location of an object in the scene is estimated by matching online measurements with the closest location feature collected a priori via offline measurements. Characteristics of RF signals which are location dependent such as Received Signal Strength Indication (RSSI) is commonly used as the feature. However RSSI estimations depend heavily on the environmental conditions, suffer from multipath effects and also exhibit nonlinear characteristics along different propagation directions. Researchers using RSSI have reported localization errors typically in the range of 3-10m [60]. The multipath effects are very predominant in indoor environments and researchers have utilized computationally intensive techniques like probabilistic methods, K-nearest-neighbor, support vector machine, etc. for localization [29].

- **Trilateration/Multilateration:** This is a range based measurement technique. Trilateration/Multilateration process determines the position of an object by measuring its distance from multiple reference points. In trilateration, the "tri" implies that at least three reference points are necessary to determine a position. Trilateration may be implemented in two steps, if both the distance and the orientation to the reference node is to be estimated: lateration and angulation. Lateration is the technique of measuring the distance between a given target and beacon. Angulation is the technique of measuring the orientation between a given target and a beacon [61].

  Lateration can be based on the measurement of the propagation time (e.g. Round-trip Time of Flight (RToF), Time of Arrival (ToA) or Time Difference of Arrival (TDoA)). In ToA, the beacon node transmits a timestamped RF signal. Target node timestamps the reception of the signal and estimates the distance with the beacon node based on the transmission time delay and speed of the transmitted signal. Time synchronization between target and beacon is a prerequisite of ToA systems, as transmission time delay has to be calculated from the transmission and reception timestamps. In TDoA, the target node estimates the distance between target and beacon node based on the TDoA between RF and ultrasound signal transmitted by the beacon node. TDoA does not require time synchronization between the target and beacon node. However, TDoA hardware is generally more complex when compared to ToA techniques. Use of TDoA is also challenging in indoor environments as ultrasonic signals are also prone to multipath transmission, reflection and scattering similar to radio frequency signals.

Angulation can be achieved using Angle of Arrival (AoA) estimation technique. The AoA technique determines the angle of arrival of the signal coming from beacon to the target [57]. Depending on whether 2D or 3D localization is desired two or three beacons are required for angulation. AOA requires highly directional antennas or antenna arrays which can measure the angles which increase the cost of the AOA system implementation. In indoor environments, AOA-based methods are affected by multipath and NLOS propagation of signals, along with reflections from walls and other objects. These factors can significantly change the direction of signal arrival and thus degrade the positioning accuracy.

2.2.4 Technologies for Indoor Localization

This section presents a review of the most prominent technologies utilized in state-of-the-art beacon based indoor localization systems. A brief review of the technologies utilized for self-localization in indoor environments is also presented.

2.2.4.1 Technologies for Beacon based Localization

The beacon messages for the various localization techniques mentioned in Section 2.2.1 can be generated based on any of the technologies mentioned in this section. The prominent categories of technologies utilized in beacon based indoor localization systems are summarized in Figure 2.2.

- **Infrared Radiation:** Indoor Localization Systems (ILS) based on Infrared Radiation (IR) utilize electromagnetic radiation with longer wavelength than visible light for positioning of objects [57]. IR based positioning requires Line-of-Sight (LOS) communication between IR transmitter and receiver. This technique is less reliable as it is prone to interference from other light sources such as fluorescent light, sunlight, etc.

- **Radio Frequency:** Radio frequency (RF) based technologies can be categorized as narrow-band technologies (eg: RFID, WLAN, Bluetooth) and wideband technologies (eg: UWB). RF technologies offer a unique advantage that LOS between beacon and target is not required for positioning [61]. The range of RF based positioning system is higher as radio waves can penetrate through obstacles. As a line of sight is not required, the Wi-Fi positioning systems have become the most widespread approach for indoor localization [62]. However, the Wi-Fi is prone to multipath effects and interference from other devices in 2.4GHz Industrial, Scientific, and Medical (ISM) radio band. This scheme requires intense site-surveying and offline calibrations as the Wi-Fi will exhibit temporal variations in signal strength along different directions.

  One of the drawbacks of using Bluetooth technology in localization is that, for each location finding process, it is required to run the device discovery procedure which significantly increases the localization latency (10–30s) and power consumption. The ZigBee technology provides a solution for short and medium range communications [63]. ZigBee based positioning is mainly implemented using scene analysis based on RSSI values.

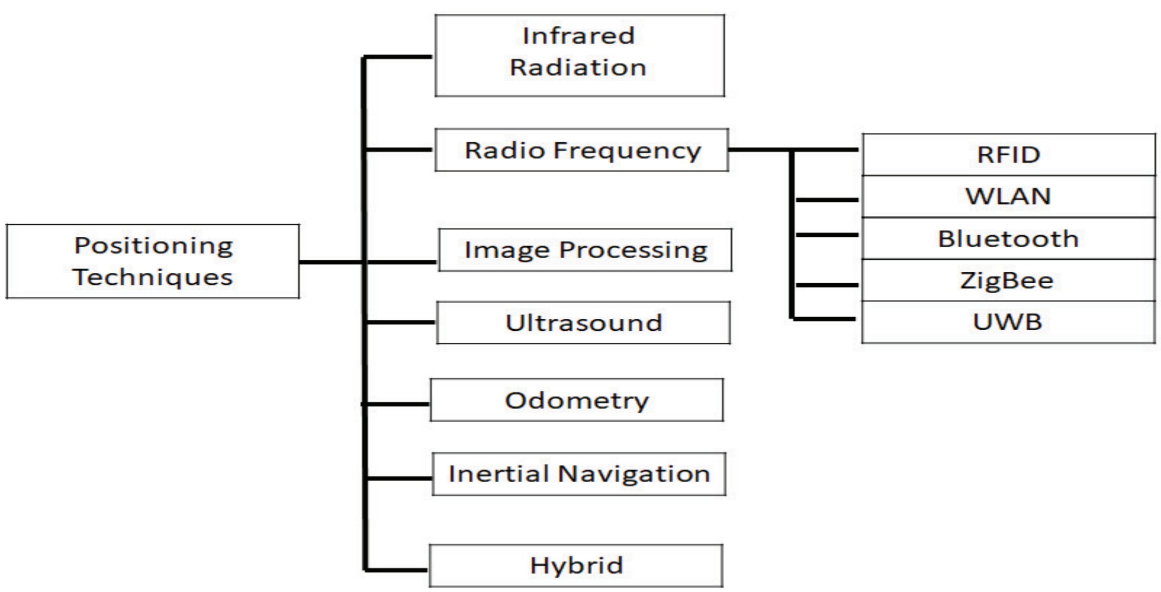

Figure 2.2. Technologies for indoor localization

Ultrawideband (UWB) is a radio technology for short-range, high-bandwidth communication which is robust to multipath effects. UWB can provide high positioning accuracy in the range of 20–30cm than which is achievable through conventional wireless technologies (RFID, WLAN etc.) [56]. However, UWB hardware is not a cost-effective solution.

- **Image processing:** Image processing based systems recognize the visual landmarks in images that are captured using high-resolution cameras. Image processing based ILS systems are most popular in robotic systems for localization in known environments. However cost incurred in the development of these systems is very high due to the cost of high-resolution cameras, memory requirements and the computational complexity of image processing algorithms. The accuracy of these systems is inadequate for environments which are not known a priori or in environments with large number of mobile objects.

- **Ultrasound:** Ultrasonic beacaon based localization offers a number of advantages over the other indoor localization systems in terms of low system cost, reliability, scalability, high energy efficiency and most importantly zero leakage between rooms. Ultrasonic signals can provide fine-grained location in the order of few 'centimeters' in Line-of-Sight conditions. However, they are prone to interference from reflected ultrasound signals propagated around by other sources, and multipath effects due to Non-Line-of-Sight condition between beacon and target nodes. Ultrasound based localization schemes are low-cost and accurate and hence considered as a suitable technology for low-cost robotic applications [64].

A comparison of the different indoor beacon based localization technologies, techniques, their typical accuracy and range of coverage is provided in Table 2.2.

2.2.4.2 Localization under NLOS Conditions

The capability to provide reasonable positioning accuracy under Non-line-of-sight (NLOS) conditions is critical for indoor beacon based localization systems. NLOS is generally described as the "Large and always positive error that arises when the distance between the transmitter and receiver is estimated from the measurement of signal transmitted between the transmitter and the receiver" [56]. As mentioned in Section 2.2.3.1, radio frequency based narrowband technologies (eg: RFID,

Table 2.2. Comparison of popular ILS technologies and techniques [59].

Localization	**Order of typical accuracy millimeter (mm)/ centimeter (cm)/ meter (m)**	**Typical range (m)**	**Localization Technique**
Camera/LIDAR	mm-m	1-10	Image processing
Infrared	cm	1-2	Thermal imaging, active beacons
Ultrasound	cm-m	3-6	TDOA
Ultra Wideband	cm-m	1-50	TOA
WLAN/Wi-Fi	m	1-5	RSSI Fingerprinting
RFID	m	1-2	Fingerprinting/Proximity detection
Bluetooth	m	2-5	RSSI fingerprinting
Zigbee	m	3-5	RSSI fingerprinting
Other Radio Frequencies	cm-m	10-1000	Fingerprinting/Proximity detection
Inertial Navigation	m	10-100	Dead Reckoning

WLAN, Bluetooth), wideband technologies (eg: UWB) or Ultrasonic signals are generally utilized as beacon signals in localization systems. NLOS is generally described as the " Large and always positive error that arises when the distance between the transmitter and receiver is estimated from the measurement of signal transmitted between the transmitter and the receiver" [56]. It is therefore critical to understand the impact of NLOS conditions on localization systems and to develop techniques that mitigate their effects.

NLOS for Ultra Wide Band (UWB) based systems is discussed in [65]. Due to the large bandwidth of UWB signals, the LOS components can be easiliy identified and extracted from the received signal. In the last few years several machine learning techniques are utilized to characterize the NLOS conditions using offline training process, followed by identification and removal of NLOS measurements during online measurements. The commonly used techniques for NLOS identification of UWB signals are hypothesis testing and machine learning, based on features from the received UWB signals [66]. Support Vector Data Description (SVDD), providing accurate data descriptions utilizing kernel techniques are proposed to perform NLOS identification in UWB positioning [67]. However the available NLOS detection systems are based on the offline learning of possible NLOS conditions and it is difficult to capture a comprehensive data set including the possible NLOS conditions in indoor environments.

Narrow bandwidth localization systems based on WLAN, Bluetooth and Wi-Fi report the Received Signal Strength Information (RSSI). The environmental factors significantly affect the RSSI characteristics, the characteristics vary significantly across different directions and as a result, the RSSI measurements are too unpredictable for range estimation [68]. Elaborate offline calibration should be adopted to reduce the influence of environment even under LOS conditions. Recently, artificial neural network based function approximation approach to map vectors of RSSI samples, known as location fingerprints, to positions on a 2D plane is proposed [69].

Ultrasound beacon based systems are preferred in low-cost systems. Development of NLOS identification and mitigation in ultrasound based localization systems is an open area of research. An ultrasound localization system which uses two different ultrasonic frequencies in Orthogonal Frequency Division Multiplexing is proposed in [64]. A Bluetooth and Ultrasound Platform for Mapping and Localization (ALPS) developed in 2015, require LOS in order to accurately determine their distances. When beacons are blocked by obstacles, the system estimates positions based on inertial data from the phone and filter out NLOS signals using an SVM classifier that checks the ratios between Bluetooth RSSI, ultrasonic signal strength and ultrasonic time of flight measurements. Even when 1200 samples of NLOS measurements were utilized for training, the system was only able to provide an accuracy of 100 cm with 80% precision. [70]. Since LOS conditions are required for accurate localization, generally large number of beacon nodes are utilized to ensure that atleast few beacons are in LOS with respect to the target node. Sufficient number of NLOS free beacon measurements are selected for location estimation based on the desired accuracy. Casas et.al utilize Least Median of Squares (LMedS) method to reject NLOS measurements when the location estimation is being carried out. [56]. Comparison of popular ILS technologies and techniques is summarized in Table 2.2. Moravek et.al have re-evaluated the 'Cricket' localization system developed in 2004 and have reported a distance estimation error of approximately 30cm for LOS condition between target and beacon code which are at a distance of 4 meters [71]. However they have not reported the location estimation accuracy of the system. From the literature survey,

Table 2.3. Performance comparison of Ultrasonic beacon based ILS systems.

ILS Systems	Accuracy	No of beacons	Technology	Consider NLOS
Caseus et.al [56]	2.5m (50% beacons in NLOS) in 10x10x5 room; ± 3 cm in LOS	8 4	Ultrasound	Yes
Moravek et.al [71]	≈ 30 cm (error in distance measurement when target and beacon node is in LOS)	1	Ultrasound	No
Lazik et.al [70]	1m- 80% precision (NLOS) 30 cm -90% precision (LOS)	3	Ultrasound Bluetooth	Yes Yes

it can be understood that cost-effective ILS solutions which do not incur higher computational and communication overhead at the same time provided reasonable accuracy under LOS and NLOS conditions are required to be developed.

2.2.4.3 Technologies for Self-localization

Self-localization of a robot can be described as the process of determination of its own position using the information available with the robot or the information collected from its onboard sensors, without using information from other reference beacons. Self-localization is the process of estimating the current position of the robot based on the last determined position and incrementing the position based on known/estimated speeds over elapsed time. While there is a need for precise self-localization, the methods suggested in the literature based on landmarks, lidars or other vision based techniques are generally expensive and require extensive computational power and system memory for performing localization in real time [4]. To make indoor autonomous robots inexpensive, it is essential to develop a self-localization method which does not require expensive sensors or memory requirements. Odometry is the most widely used method for determining the momentary position of a mobile robot in which motion sensors are used to estimate the change in position over time. The fundamental principle of odometry or dead-reckoning is to integrate incremental motion information over time. The most commonly used motion sensors are position/rotary encoders attached to the wheel of the robot. The common errors in odometry based localization can be classified as (1) systematic errors- caused by kinematic imperfections of the mobile robot (e.g. unequal wheel diameters, or orientation of wheel encoder with respect to the wheel), and (2) non-systematic errors- caused by wheel slippage or irregularities of the floor. Systematic errors arise due to the unavoidable imperfections in the design of the robot and these errors are relatively constant over prolonged periods of time, while non- systematic errors are due to parameters external to the robot like irregularities on the floor and are non-deterministic. Even if the position or rotary sensors are inexpensive, the accuracy of these sensors is severely limited in detecting the precise rotations or to distinguish the slip of robot from the rotation of robot around its own axis. Odometry based techniques lead to unbounded accumulation of errors over a period of time. However, once a precise system error model and its parameters are given, the accuracy of odometry can be remarkably improved by using suitable control techniques based on the kinematics of the robot is to be developed [72–76].

Self-localization is critical in MRS and more importantly in SRS since robots in swarm are more cooperative in nature and may even aggregate together for performing certain tasks. Hence the robots should be able to move towards the target or other robot using its onboard sensors. With the advancements in MEMS technology, the inertial measurements units (IMU) can serve as a suitable substitute for motion sensor in place of position or wheel encoders [77]. An IMU generally consists of a two or three-axis accelerometer, gyroscope, magnetometer and optionally a temperature sensor or a digital signal processor. The major advantage of utilizing IMU based localization is that a localization scheme independent of the robot structure can be developed using IMU. IMU based techniques can also be utilized for 3D localization of robot or for other types of robot including legged robots or even can be used for tracking of human beings in indoor environments [78].

2.3 Task Allocation of MRS and SRS Systems

‘Task allocation’ is to decide “which robot executes which task, at what time”. The problem of distributing and scheduling a set of tasks among a group of robots to achieve the system goals taking into account the operational constraints is described as multi-robot task allocation (MRTA) problem. The first reported work on the formal classification of MRTA problem was by Gerkey and Matarić in 2004 as depicted in Figure 2.3 [31]. They proposed a taxonomy for the classification of MRTA problem along three axes-tasks, robots and assignment as follows:

- Single-Task robots (ST) Versus Multi-Task robots (MT)
- Single-Robot tasks (SR) Versus Multi-Robot tasks (MR)
- Instantaneous Assignment (IA) Versus Time-extended Assignment (TA)

The ST/MT classification is based on the number of tasks that can be assigned to a robot in each task assignment cycle. ST robots can be assigned (or can complete) only a single task whereas MT robots can be assigned multiple tasks in a task assignment cycle. SR/MR classification is based on the number of robots required to complete a task. SR tasks can be completed by a single robot whereas MR tasks require multiple robots to complete the task. The IA schemes consider only instantaneous conditions of the system, i.e the robot, task and the current conditions of the environment for task allocation, whereas TA schemes perform task allocation by taking into account, current and future requirements of the system. ST-SR-IA is the simplest task allocation problem to solve in which each robot is assigned a single task and each task is attended by only one robot. MT-MR-TA is the most complex among all MRTA problems in which each robot is assigned multiple tasks considering the current and future requirements and each task requires multiple co-operating robots for its completion.

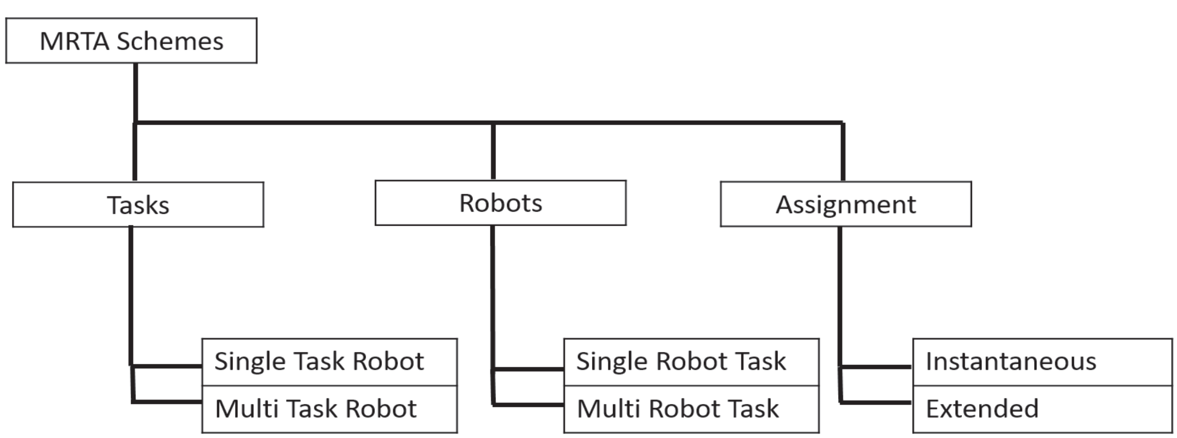

Figure 2.3. MRTA Taxonomy proposed by Gerkey and Matarić

The taxonomy, iTax proposed in 2013 by G.Korsah et al. is an extension to the taxonomy proposed by Gerkey and Matarić [79]. G.Korsah et al. identify that a key distinguishing factor between different types of MRTA problems is the degree of interdependence of robot-task utilities in the task assignment problem. In MRTA systems, the utility functions indicate, how the robots skills can match the tasks requirements. Gerkey and Matarić, defines the utility of a robot for a particular task as follows: Given a robot 'R' and a task 'T', if 'R' is capable of executing task 'T' with quality Q_{RT} and cost C_{RT}, then the utility measure, U_{RT} is given by

$$U_{RT} = Q_{RT} - C_{RT}; \tag{2.6}$$

If 'R' is not capable of executing task 'T' then the utility measure, U_{RT} is given by

$$U_{RT} = -\infty; \tag{2.7}$$

Task allocation, if formulated as an optimization problem, seeks to determine a feasible assignment of tasks to robot which optimizes certain objectives(utility function). G.Korsah et.al. proposed a two-level taxonomy in which the first level defines the degree of interdependence of robot-task utilities. The taxonomy proposed by Gerkey and Matarić is further extended by G.Korsah et.al. based on utility classification as indicated in Figure 2.4. The second level extend the classification of Gerkey and Matarić by adding further descriptive information about the robot-task utilities. Based on the degree of interdependency four possible classifications were proposed as follows. 1) no dependencies: the utility of robot-task does not depend on the task of any other robot or its own schedule, 2) in-schedule dependencies: the utility of robot–task depends on its own schedule, 3) cross-schedule dependencies: the utility of robot–task depends on its own schedule and the schedules of other robots, 4) complex dependencies: the utility of robot–task depends not only the schedules of other robots but also on their decomposition.

2.3.1 Different Approaches for Solving MRTA Problem

The solutions suggested for MRTA problem in literature can be classified into two major categories- optimization based approaches and market based approaches. Optimization based approaches attempt to find a task allocation such that, for the given union of state of the system, it is impossible to

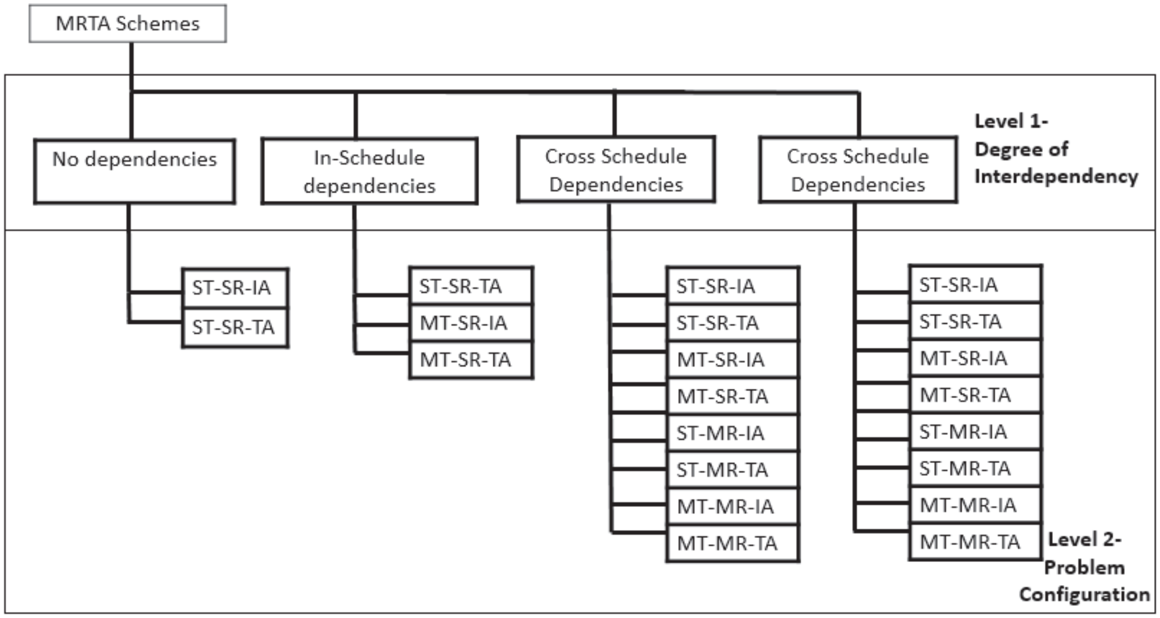

Figure 2.4. iTax MRTA Taxonomy.

find a better allocation than the obtained one. When treated as an optimization problem, out of the eight classifications proposed by Gerkey and Matarić only the ST-SR-IA problem can be solved in polynomial time, while remaining problems are strongly NP (Non-deterministic Polynomial time) hard [31]. Similarly, only problems in 'no dependencies' category can be solved in polynomial time, while the other problems are NP-hard if treated as an optimization problem. Different optimization based approaches are summarized in Figure 2.5.

Population based approaches like genetic algorithm or heuristic approaches like simulated annealing [80, 81] are proposed to find the optimal allocation of robots to tasks such that the cost function is minimized and the utility is maximized. The most common cost functions utilized are the distance traveled by the robot, the time taken to travel the required distance etc. Although in static environments and ideal scenarios, the distance to be traveled and the time required to cover a distance by the robot may be calculated, the assessment of the same is not a realistic in case of dynamic environments or unmapped terrains. The optimization based approaches are designed to solve one of the 8 possible combinations of MRTA as suggested by Gerkey and Matarić. The SR-ST-IA problem is the most commonly addressed problem in literature. Existing solutions combine the path planning and navigation with task allocation and hence the method cannot be used for different kinds of robots such as wheeled, crawling or biped robots.

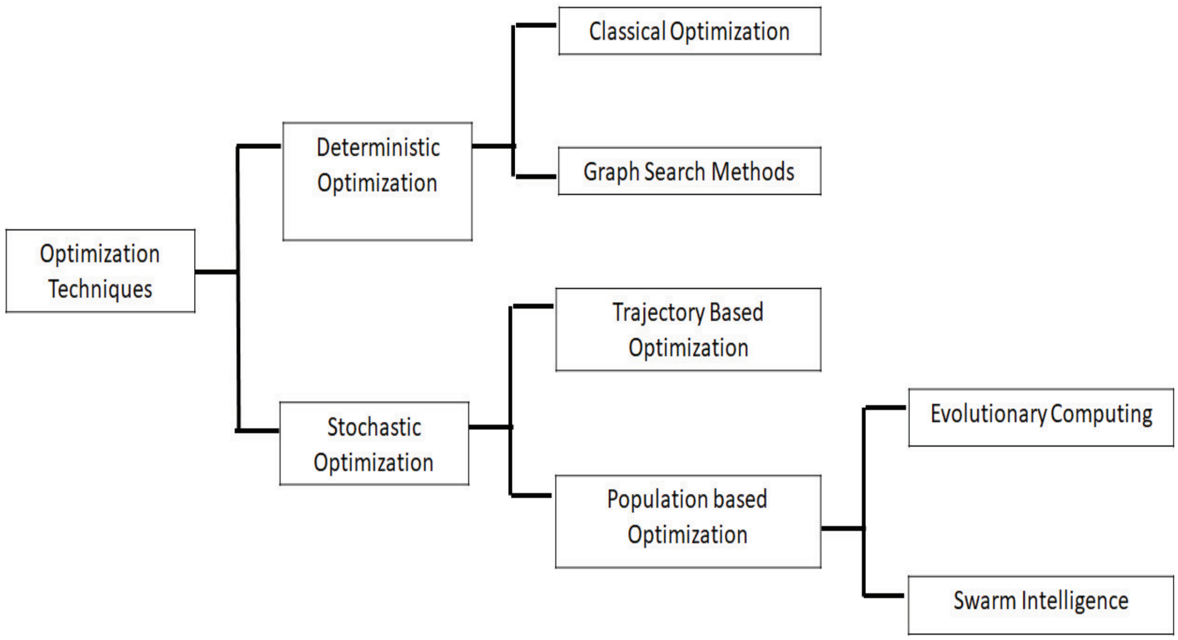

Figure 2.5. Optimization based Approaches

The optimization based approaches are difficult to implement for practical scenarios due to it computational and communication complexity. For example, for an SR-ST problem with 'm' robots and 'n' tasks, since each robot must inform other robots about its own utility of the tasks, the broadcast overhead is $O(n)$/robot. If we assume that if each robot can find the optimal solution, it needs to perform 'm' comparisons to find the best utility among the 'm' robots available for each task. Thus, the overall computational and communication complexity is $O(mn)$ for each robot.

Market based approaches for task allocation is emerging as a popular choice within robotics research community in recent years. In market based approaches instead of finding the optimal solution, focus is on finding a feasible solution as fast as possible. In market-based approaches the task allocation is based on a bidding-auctioning procedure between a central agent (auctioneer) and the robots (bidders). The auctioneer announces the task along with the information pertaining the task such as, the number of robots required to perform the task, energy budget for the task, capabilities of the robot to service the task, deadline of the task etc. The robots which listen to the auction announcement will then bid for the tasks. Based on how the winner of the bidding is declared the market based approaches can be centralized or distributed [82, 83]. Most commonly, the central auctioneer will evaluate the bids and assign the task to the robot with the best bid. The communication overhead increases drastically in these scenarios when the size of the network increases. In-addition to this, fully centralized approaches can be computationally intractable and unresponsive to change. However, task allocation problem is a dynamic decision problem, that needs to be iteratively reconsidered over time rather than as a static assignment problem. From the literature review it is concluded that there is a need to to develop a task allocation scheme for SRS and MRS system with the following characteristics:

1. Most of the solutions available for task allocation in literature is limited to allocating a single robot to a single task which is the simplest of the task allocation problems. However, with the increased cooperative behaviour offered by SRS, it is required that the task allocation protocol can allocate the task to the robots, belonging to either of the eight categories of MRTA identified by Gerkey and Matarić depedending on the application requirement.

2. Instead of considering the task allocation as a static assignment problem, the task allocation should be treated as a dynamic decision problem which will be responsive to variation in environmental conditions or scaling population of robots.

3. The task allocation scheme should be independent of path planning or navigation of robots, so that the scheme can be used for task allocation of land/air-borne robots or for co-operating networks in land and air.

4. The scheme should identify the dependencies with the other layers of the protocol stack so that the same can be incorporated into any existing robotic system.

5. The scheme should be scalable at the same time computationally efficient with lesser communication overhead

2.4 Conclusions

The major research gaps related to time synchronization, localization and task allocation in SRS and low-cost MRS identified based on literature review are listed below.

- Based on the literature review, it is concluded that a distributed, global time synchronization protocol suitable for mobile, multi-hop and scalable network like a swarm of robots or multi-robotic systems is not available in the literature. Most of the protocols available in literature suggest a resynchronization interval in the order of few seconds as mentioned in Table 2.1. The wireless channel will be utilized by the robot for several other functionalities like localization, cooperative path planning, navigation, aggregation, dispersion, task allocation, task monitoring, etc. If the resynchronization interval is in the order of few seconds, then the robots will not be able to use the channel effectively for other functionalities. Hence, an improvement in the resynchronization interval is essential. It is essential that the synchronization error of all nodes within the network is deterministic and bounded to facilitate the development of other layers of protocol stack such as MAC and Routing, localization and task allocation of robots. To maximize the utilization of robots, they should be allowed to move seamlessly across clusters without necessitating resynchronization or introducing overheads in synchronization. Hence a topology independent time synchronization protocol is to be developed. The protocol should also be scalable such that addition or deletion of few nodes should not necessitate resynchronization of network or drastically affect the accuracy of networkwide time synchronization.

- In indoor environments, the beacon signals are prone to Non-Line-of-Sight (NLOS) conditions due to the presence of obstacles and other moving objects. The beacon signals may also get reflected and scattered leading to multipath effects. It is therefore critical to understand the impact of NLOS conditions on the beacon signal and to develop techniques that mitigate their effects. Although ultrasonic beacon based localization systems are preferred in low-cost SRS and MRS, the Indoor Localization Systems (ILS) which utilize ultrasonic beacons provide an accuracy in the order of approximately 1-2.5 meters under NLOS conditions while utilizing 3-8 beacons as described in Table 2.3. Hence, there is a need for developing a low-cost indoor localization system which can provide an accuracy of few tenths of centimetres under LOS and NLOS conditions. The suggested localization scheme should require lesser number of beacons to reduce the cost of deployment.

- For robots in SRS and MRS, operating in GPS-denied indoor environments, relative localization is an essential capability required by the robots for cooperative task completion. The robots should also be able to self-localize using their on onboard sensors so that the communication between the robot and the beacons for localization can be minimized.

- Most of the solutions available for task allocation in literature is limited to allocating a single robot to a single task which is the simplest of the task allocation problems. However, with the increased cooperative behaviour offered by SRS, it is required that the task allocation protocol can allocate the task to the robots, belonging to either of the eight categories of MRTA identified by Gerkey and Matarić depedending on the application requirement. Hence, generic task allocation framework which can be utilized to solve any of the eight types of MRTA problem needs to be developed. A reactive task allocation scheme which considers task allocation as a dynamic decision problem, that needs to be iteratively reconsidered over time rather than as a static assignment problem is required for SRS ans MRS. The task allocation scheme should be scalable and computationally efficient which incurs minimum communication overhead.

- Most of the works reported on time synchronization, localization and task allocation do not cite their dependencies with other layers of protocol stack or hardware requirements of implementation of the same, thus making the implementation of the protocol difficult for practical purposes. It is important that the suggested solutions clearly define its dependency with the other layers of the protocol stack so that the hardware, communication and the interoperability of the different services and other layers of the protocol stack are well established.