Heterogeneous Bayesian Decentralized Data Fusion: An Empirical Study

Ofer Dagan, Tycho L. Cinquini and Nisar R. Ahmed

Abstract—In multi-robot applications, inference over large state spaces can often be divided into smaller overlapping sub-problems that can then be collaboratively solved in parallel over ‘separate’ subsets of states. To this end, the factor graph decentralized data fusion (FG-DDF) framework was developed to analyze and exploit conditional independence in heterogeneous Bayesian decentralized fusion problems, in which robots update and fuse pdfs over different locally overlapping random states. This allows robots to efficiently use smaller probabilistic models and scalably fuse relevant local parts of a larger global joint state pdf, while accounting for data dependencies between robots. Whereas prior work required limiting assumptions about network connectivity and model linearity, this paper relaxes these to empirically explore the applicability and robustness of FG-DDF in more general settings. We develop a new heterogeneous fusion rule which generalizes the homogeneous covariance intersection algorithm, and test it in multi-robot tracking and localization scenarios with non-linear motion/observation models under communication dropout. Simulation and linear hardware experiments show that, in practice, the FG-DDF continues to provide consistent filtered estimates under these more practical operating conditions, while reducing computation and communication costs by more than 95%, thus enabling the design of scalable real-world multi-robot systems.

I. INTRODUCTION

Bayesian decentralized data fusion (DDF) [1] is applicable to networks of robots acting in a shared (problem) space toward common goals that require estimation over a global set of random variables (rvs). Robots can gain new data from local sensors and by peer-to-peer communication of their current local estimated joint probability distribution function (pdf), often described only by their mean and covariance over the full, homogeneous, set of rvs. Heterogeneous DDF is the sub-class of DDF problems where communicating robots fuse information with respect to pdfs over two non-equal, but overlapping, subsets of the rvs [2].

Many collaborative applications across robotics are instances of heterogeneous fusion. Heterogeneous fusion enables scalable operation of large robotic teams by distributing the global joint inference problem to smaller, overlapping, local ones. Thus robots are able to reason over their local inference task and communicate only relevant data to their neighbors. For example: (i) in multi-robot simultaneous localization and mapping (SLAM) [3], robots keep their estimated positions local, and share only parts of the map; (ii) in multi-robot tracking with sensor bias uncertainties, robots can share estimates over common targets, while non-mutual targets and sensor biases are only estimated locally [2]; (iii) when estimating local sensor measurement bias and the temperature distribution across a room, then bias estimates are kept local while temperature estimates are shared [4]. Thus, enabling each robot to reason over and communicate only parts of the full joint inference problem, is imperative for scalability, as local communication and computation requirements now scale with each robot’s local inference task and not with the full global inference problem. Previous work [2],[5], shows that by exploiting the natural sparse structure inherent in robotic heterogeneous fusion problems, communication and computation costs can be reduced by more than 95%.

One of the main challenges in DDF is to correctly account for dependencies in the data gathered and shared by robots, so that new data is treated as new only once. In classical homogeneous DDF, where robots communicate and infer the same global set of rvs, methods exist to either explicitly (e.g., by using a channel filter (CF) [6]) or implicitly (e.g., covariance intersection (CI) [7], inverse covariance intersection (ICI) [8]) track such dependencies. In heterogeneous DDF, the problem becomes more acute, as in addition to the above known-unknown dependencies, there are now (hidden) unknown-unknown dependencies between non-mutual rvs that must be treated [9]. To study, analyze and solve heterogeneous DDF problems and the dependencies therein, a factor graph based framework, dubbed FG-DDF, based on the heterogeneous state CF (HS-CF) algorithm [2], was developed in [5] and then extended to linear-dynamic systems in [9]. However, some fundamental issues arise for non-linear systems – namely, the definition and meaning of common data dependencies become less clear, e.g. when robots propagate their pdfs using linearization based on different current estimates. Also, it is neither obvious nor clear what impact real-world issues such as imperfect communication (message dropouts) have on heterogeneous DDF, since these can lead to robots having different perspectives on common data dependencies.

This paper empirically explores the robustness and applicability of the FG-DDF framework in realistic heterogeneous multi-robot scenarios. We show that although it is not theoretically clear how common data dependencies are exactly tracked in non-linear systems, in practice, the FG-DDF framework yields consistent estimates at each robot, even when 50% of the messages do not get to their destination. We extend FG-DDF by developing a new heterogeneous CI
TABLE I: Key notations and definitions used in the paper

| Symbol | Description |
|--------|-------------|
| \(N_r\) | Set of \(n_r\) robots |
| \(N_i^j\) | Set of robot \(i\)'s (network) neighbors |
| \(\chi_i\) | Robot \(i\)'s set of rvs |
| \(\chi_i^L\) | Local rvs, only monitored by \(i\) |
| \(\chi_i^C\) | Set of rvs common to \(i\) and \(j\) |
| \(\chi_i^C\cap\{\chi_j^L\}\) | Non-mutual rvs to \(i\) and \(j\) |
| \(\chi_i^C\times\chi_j^L\) | \(i\)'s available data at time \(k\) prior to fusion |
| \(\chi_i^C\times\chi_j^L\) | \(i\)'s available data at time \(k\) post fusion |
| \(Z_i^{c}\) | \(i\)'s common to \(\chi_i^C\) and \(\chi_j^L\) |
| \(Z_j^{c}\) | \(j\)'s available data at time \(k\) |
| \(Z_k^{c}\) | \(k\)'s available data at time \(k\) |

The rest of the paper is organized as follows. Sec. II defines the heterogeneous fusion problem and reviews related work. Sec. III presents the technical approach for nonlinear heterogeneous DDF and develops a new heterogeneous CI rule. Sec. IV details the empirical multi-robot simulation and hardware studies. Sec. V summarizes the findings and describes future work.

II. PROBLEM STATEMENT

Consider a network of \(|N_r| = n_r\) autonomous robots, jointly monitoring a global set of rvs \(\chi_k\), where each robot \(i \in N_r\) is tasked with inferring (estimating) an overlapping subset of (dynamic) rvs \(\chi_k \subset \chi_k\). Each robot recursively updates its local prior pdf over \(\chi_k\) in four main steps: (i) prediction, using the conditional transition probability \(p(\chi_k^t|\chi_k^{-1})\); (ii) marginalization of rvs from past time step; (iii) Bayesian fusion of local sensor data \(y_i^t\), described by the conditional likelihood \(p(y_i^t|\chi_k^t)\); and (iv) fusion for any neighboring robot \(j \in N_i^j\) by exchanging pdfs over their common rvs via the peer-to-peer heterogeneous State (HS) fusion rule [2],

\[
p_f^i(\chi_i^t|Z_k^{c,+}) \propto \frac{p_f^i(\chi_i^t|Z_k^{c,-})}{p_i(\chi_i^t|Z_k^{c,-} \cap Z_k^{c,+})} \cdot p_i(\chi_i^t|Z_k^{c,+}),
\]

where \(p_f^i(\cdot)\) is the fused posterior pdf at robot \(i\), \(p_i(\cdot|Z_k^{c,-} \cap Z_k^{c,+})\) is the pdf over robots \(i\) and \(j\) common rvs, given their common data, which can stem from common prior, dynamic models, and previous communication episodes. The rest of the notation is defined in Table I. Note that when \(\chi_i^t = \chi_i^C = \chi\), (1) degenerates to the homogeneous Bayesian fusion rule [1].

Previous work [2] shows that (1) is only valid when non-mutual variables are conditionally independent given common variables between the communicating robots, i.e., \(\chi_i^C \independent \chi_j^L|\chi_i^C\). Later work in [5] develops a factor graph-based framework as both the inference engine and as a tool track dependencies in the data, analyze and exploit conditional independence structure in heterogeneous fusion problems. This is then used in [9] to develop a method for conservative filtering for heterogeneous fusion in dynamic systems. Since the aforementioned body of work aimed at gaining fundamental understanding of the heterogeneous DDF problem and the nature of dependencies in the data held by the robots in the network, several assumptions were made: (i) the dynamic system transition and observation models \((p(x_i^{t+1}|x_i^t))\) and \(p(y_i^t|x_i^t)\), respectively are linear with additive white Gaussian noise (AWGN); and (ii) the network communication topology is described by an undirected cyclic graph.

However, questions arise as to how robust/applicable the FG-DDF framework developed in [5] and [9] is to: nonlinear transition and observation models; real-world problems such as message dropouts; and approximations of the ‘common’ pdf, \(p_f^i(\chi_i^t|Z_k^{c,+} \cap Z_k^{c,-})\), e.g., via a heterogeneous version of the covariance intersection (CI) algorithm. This paper explores those questions by an empirical study using simulations and hardware experiments.

Related work: In heterogeneous fusion problems, marginalization often couples previously conditionally independent rvs. Since maintaining conditional independence is key to the solution approach, we identified two main aspects of the problem that affect the solution: (i) type of common \((\chi_i^C)\) and local variables \((\chi_j^L)\), i.e., whether they are dynamic or static; (ii) whether the inference algorithm solves for static variables, dynamic variables with a smoothing approach, or dynamic variables with a filtering approach.

In [4], Paskin and Guerstrin describe a distributed junction tree (D-JT) algorithm to infer the temperature field in a lab setting, in the presence of local sensor measurement bias. Here all variables are static, and sensors estimate and share a subset of static temperature variables, while keeping their bias estimates local. Thus, conditional independence structure in this case is not affected by marginalization and stays constant in time. Makarenko et al. [10] extend the D-JT to include dynamics and cast it as a DDF problem. But their algorithm is then only applied to a single variable of interest, i.e., it describes a homogeneous fusion problem, which does not necessitate maintaining a conditional independence structure. In [3], Cunningham et al. develop a smoothing and mapping (SAM) technique called DDF-SAM, based on factor graphs. As their work focuses on SAM, the shared variables are static (subset of the map) and the solution (which includes the dynamic robot states) is smoother-based, i.e., the algorithm circumvents the challenges resulting from marginalizing past states, as the map variables are
independent given the full robot trajectory. Chong and Mori [11] use information graphs and Bayes nets to analyze and design algorithms for nonlinear distributed estimation. Their work presents a wide analysis of the problem to maintain conditional independence in the dynamic case, but assumes a deterministic state process, and does not account for stochastic dynamic problems.

From the heterogeneous fusion perspective, a filtering solution to a stochastic dynamic system (with dynamic local and common variables) is a more general and challenging scenario as it becomes harder to: (i) correctly remove common data in fusion, as it is ‘rolled up’ into the current estimate upon marginalization of past states (especially when it was propagated through a non-linear transformation), and (ii) maintain conditional independence between non-mutual states. For these reasons, this paper uses the combined problems of dynamic multi-target tracking and self-localization as a test case to explore the heterogeneous FG-DDF framework. In this scenario, robots independently localize themselves based on range and bearing measurements to known landmarks, while sharing state estimates on common tracked dynamic targets.

III. TECHNICAL APPROACH

There are two key points pertaining to the heterogeneous fusion rule in [1]. First, to guarantee its validity, conditional independence between non-mutual rvs must be exploited and maintained. Here we use the FG-DDF framework developed in [5] as the local inference engine at every robot. The second point – which lies at the core of DDF problems in general and heterogeneous DDF specifically—he is how to account for dependencies in the data shared between robots, so that new data are treated as such only once. In heterogeneous DDF, these dependencies are accounted for in the ‘common’ pdf \( p^j_C(x_C; \chi_C \cap Z_j^{-}) \), in the denominator of [1]. The DDF literature distinguishes between exact methods, where dependencies are explicitly tracked either by: (i) keeping a pedigree of data transition in the network [12] or adding a channel filter (CF) [6]; or (ii) approximate methods such as covariance intersection (CI) [7] and inverse covariance intersection (ICI) [8], where unknown dependencies between the estimates are removed at the cost of inflating the covariance matrix. Below we provide a summary of the main technical details of the CF and CI methods, and how they are used for heterogeneous DDF. These methods will be used to empirically study the FG-DDF framework in dynamic nonlinear systems.

A. Factor Graphs for DDF (FG-DDF)

A factor graph is a type of probability graphical model (PGM) [13], which has gained popularity in the robotics community since it naturally expresses the sparse information dependency structure inherent in many robotic applications [14]. Recent work used factor graphs to exploit conditional independence in heterogeneous DDF problems and suggested a new framework, FG-DDF, to analyze and solve them in linear static [5] and dynamic [9] systems.

Briefly, a factor graph is an undirected bipartite graph \( \mathcal{G} = (F, V, E) \), factorized into smaller functions given by factor nodes \( f_i \in F \). Each factor \( f_i(V_i) \) is connected by edges \( e_{lm} \in E \) only to the function’s random variables \( v_m \in V_i \subset V \). The joint distribution over the graph is then proportional to the global function \( g(V) \).

\[
p(V) \propto g(V) = \prod_i f_i(V_i). \tag{2}
\]

In FG-DDF [5], new factors are added to the graph either due to prediction, observation, and fusion. In filtering, the graph is also manipulated or re-factorized to maintain the conditional independence structure and ensure the estimate is conservative [9]. Since these works assumed linear models and Gaussian noise, each factor was expressed using the information (canonical) form of the Gaussian distribution, i.e., \( f_i(V_i) \propto \mathcal{N}(\xi_i, \Lambda_i) \), where \( \xi_i \) and \( \Lambda_i \) are the information vector and matrix, respectively. This paper explores the heterogeneous DDF problem in non-linear systems using an augmented version of the extended information filter (EIF) [15]. The EIF works by propagating the information vector and matrix of the prior Gaussian distribution through a linearized model, thus factors are still expressed in the same way as in the linear FG-DDF.

B. Channel Filter

In networks with an undirected a-cyclic communication graph, there is only one communication path between any two robots. For such networks, [6] suggests adding a filter on the communication channel between every pair of communicating robots, \( i \) and \( j \), to explicitly calculate \( p^j_C(x_C|Z_k^{-} \cap Z_j^{-}) \) over the full (homogeneous) set of rvs \( \chi_C \). Reference [2] extends this idea to heterogeneous DDF with the HS-CF. In HS-CF, the CF recursively computes the marginal pdf \( p^j_C(x_C|Z_k^{-} \cap Z_j^{-}) \) in [1], which is then removed from the robots’ communicated marginal pdfs. When the pdfs are expressed using the marginal information vector (\( \xi_C \)) and matrix (\( \Lambda_C \)), representing the mean and covariance of the pdf, the fusion over the subset of common rvs \( \chi_C \), shown in the left part of [1] is,

\[
\begin{align*}
\bar{\xi} & = \bar{\xi} + \xi_C \Lambda_C^{-1} \xi_C \bar{\xi} + \xi_C \Lambda_C^{-1} f_c \xi_C \bar{\xi} + \Lambda_C^{-1} f_c \xi_C \Lambda_C^{-1} f_c \xi_C \bar{\xi} \\
\bar{\Lambda} & = \bar{\Lambda} + \xi_C \Lambda_C^{-1} \xi_C \Lambda_C^{-1} f_c \Lambda_C^{-1} \xi_C \Lambda_C^{-1} f_c \xi_C \Lambda_C^{-1} \xi_C \Lambda_C^{-1} f_c \xi_C \Lambda_C^{-1} \xi_C \Lambda_C^{-1} f_c \\
& \quad + \Lambda_C^{-1} f_c \Lambda_C^{-1} \xi_C \Lambda_C^{-1} f_c \Lambda_C^{-1} f_c \xi_C \Lambda_C^{-1} \xi_C \Lambda_C^{-1} f_c \\
& \quad + \Lambda_C^{-1} f_c \Lambda_C^{-1} f_c \Lambda_C^{-1} f_c \Lambda_C^{-1} f_c \
\end{align*}
\tag{3}
\]

In practice, each robot maintains another factor graph on every communication channel, representing the CF with its neighboring robots. Note that in a non-linear system where homogeneous DDF is used (i.e. all robots estimate the same state vectors), there are no guarantees that the CF computes the exact pdf over the common data, as in many cases it is propagated forward via linearization, using different linearization points for each robot. Heterogeneous fusion might be more sensitive due to that fact the linearization points are guaranteed to be different, as the HS-CF holds only data about the marginal pdf over the common state variables for each robot.
C. Covariance Intersection

Covariance intersection (CI) [7] is a widely used approximate method for cyclic or ad-hoc communication topologies. CI computes the weighted average of the robots’ information vector and matrix, where the weight, \( \omega \), is calculated to optimize some predetermined cost function, e.g., the determinant or trace of the fused covariance matrix. The CI fusion rule is then given by,

\[
\begin{align*}
\zeta^i_f &= \omega \zeta^i + (1 - \omega) \zeta^j + \left[ (1 - \omega) \zeta^i + \omega \zeta^j \right] , \\
\Lambda^i_f &= \omega \Lambda^i + (1 - \omega) \Lambda^j + \left[ (1 - \omega) \Lambda^i + \omega \Lambda^j \right], \\
\end{align*}
\]

where we show that the information vector and matrix of the ‘common’ pdf \( p^i_j(x_i|x_c) \cap p^j_i(x_j|x_c) \) can be approximately evaluated using \( \omega \). With this interpretation, we can replace the CI calculated \( \zeta^{ij}_{x^c} \) and \( \Lambda^{ij}_{x^c} \) in (3) with,

\[
\begin{align*}
\zeta^{ij}_{x^c} &= (1 - \omega) \zeta^{i}_{x^c} + \omega \zeta^{j}_{x^c} , \\
\Lambda^{ij}_{x^c} &= (1 - \omega) \Lambda^{i}_{x^c} + \omega \Lambda^{j}_{x^c} .
\end{align*}
\]

The HS-fusion rule in [1], when used with the above expression to approximate the first two moments of the denominator, is then dubbed the HS-CI fusion rule.

IV. EMPIRICAL STUDY

Our empirical study aims at testing the robustness and applicability of FG-DDF to realistic heterogeneous data fusion challenges. Using Monte Carlo simulation, we test the applicability of the explicit (HS-CF) and implicit (HS-CI) heterogeneous fusion rules for a network of robots performing non-linear dynamic multi-target tracking and self-localization. We then test robustness to message dropouts in both simulation and linear hardware experiments.

A. Simulation Description

We simulated a team of \( n_r = 5 \) robots, tracking \( n_t = 6 \) targets, connected in an undirected chain network (1 \( \leftrightarrow \) 2 \( \leftrightarrow \) 3 \( \leftrightarrow \) 4 \( \leftrightarrow \) 5). Note that a chain network is the worst case for an a-cyclic topology (e.g., a tree), as it takes more time to propagate to the ends of the network [6]. The global inference task of the team of robots is to infer the dynamic target positions, \( r^i_k = [x^i_k, y^i_k, \theta^i_k]^T \) of all robots \( i \in N_r \), and 2D dynamic target positions, \( r^m_k = [x^m_k, y^m_k]^T \), of all targets \( m = 1, 2, ..., n_t \). The individual inference task of each robot is given in [6], with \( x^i \) denoting the vector of rvs (states) that each robot \( i \) infers (estimates).

\[
\begin{align*}
\chi^1_k &= \begin{bmatrix} r^1_k \\ t^1_k \\ \ell^1_k \end{bmatrix} , & \chi^2_k &= \begin{bmatrix} r^2_k \\ t^2_k \\ \ell^2_k \end{bmatrix} , & \chi^3_k &= \begin{bmatrix} r^3_k \\ t^3_k \\ \ell^3_k \end{bmatrix} ,
\end{align*}
\]

It can be seen that at maximum, a robot estimates 9 states (robot 3) and communicates 4 (robots 3-4). Compared to homogeneous DDF over the global 27 states vector, this translates to 99% communication and 96% computation reduction.

At every time step, robots: (i) take local sensor bearing and range measurements with respect to maximum 4 known landmarks (to localize themselves) and with respect to their perspective targets; (ii) communicate factors via heterogeneous FG-DDF over subsets of common target variables \( x^i \) with their neighbors. For example, common variables for robots 3 and 4 are \( x^{34} = x^3 \cap x^4 = [t^3_k, t^4_k]^T \). Note that robots do not take relative measurements to each other (as in cooperative localization), but their own position estimates will nevertheless get indirectly updated due to dependencies on common target positions.

The robots’ follow nonlinear Dubin’s cars dynamics,

\[
\begin{align*}
\dot{x}^i &= v^i \cos \theta^i + \omega^i , \\
\dot{y}^i &= v^i \sin \theta^i + \omega^i , \\
\dot{\theta}^i &= \frac{v}{L} \tan \phi^i + \omega^i ,
\end{align*}
\]

where \( v^i \) and \( \phi^i \), and \( \omega^i = [\omega^i_x, \omega^i_y, \omega^i_\theta]^T \) are the time dependent linear velocity (m/s), steering angle (rad), and zero mean additive white Gaussian noise (AWGN) of robot \( i \), respectively. \( L \) is the front-rear wheel distance (taken to be 0.6m in the simulations). Target \( m \)’s linear dynamics is modeled with an assumed known motion control law,

\[
\begin{align*}
x^{m}_{k+1} &= x^{m}_{k} + u^{m}_{x,k} + \omega^{m}_{x} , \\
y^{m}_{k+1} &= y^{m}_{k} + u^{m}_{y,k} + \omega^{m}_{y} ,
\end{align*}
\]

where \( u^{m}_{k} = [u^{m}_{x,k}, u^{m}_{y,k}]^T \) is the motion control input, and \( \omega^{m} = [\omega^{m}_{x}, \omega^{m}_{y}]^T \) is again zero mean AWGN. All robots were initially randomly positioned in \( 20m \times 20m \) square and then normally sampled in each simulation with \( \sigma^2 = 25m^2 \), and target positions were randomly sampled from a normal distribution with \( \sigma^2 = 2m^2 \). Known landmarks were positioned in a \( 200m \times 200m \) square.

Implicit vs. Explicit Data Tracking: The first set of simulations tests the heterogeneous FG-DDF framework to non-linear dynamics and measurement models. We performed 50 MC simulations using the HS-CF and HS-CI to explicitly and implicitly account for common data dependencies, respectively. These dependencies can arise due to both robots using the same target dynamic model, and from previous communication episodes. As they are non-linearly propagated in time and ‘rolled up’ into the current estimate upon filtering, it becomes harder to correctly remove common information during fusion according to [1].

The results in Fig. (a) compare the average root mean squared error (RMSE) and 2\( \sigma \) bounds for estimates of the local state vectors in [6] across 50 MC runs relative to the marginal over each robot’s subset of states as computed from a centralized EIF, i.e. estimating the mean and covariance of

\[1\]While the NEES chi-square consistency test [16], [17] is a more indicative test for consistency, here we show the mean squared error, as it is visually clearer and simpler. Nevertheless, we confirmed consistency via the NEES test with results being consistent 80% – 99% of the time, depending on the robot.
The full joint 27 random state vector $\chi_k = \bigcup_{i \in N_r} \chi^i_k$. We show results of robots 1, 3, and 3, which are the ‘end of the chain’ and middle robots, respectively (the results of robots 2 and 4 are qualitatively similar to 1 and 3).

As seen from the figure, all robots yield a good estimate with an average RMSE that is smaller than the average $2\sigma$ bounds for all time steps, with some larger error ‘jumps’ which we attribute to observability loss for robot pose or target states (i.e. poor sensing geometry). As expected, both heterogeneous fusion rules result in a larger error and larger variance with respect to the centralized estimator; this is the result of the conservative filtering algorithm [9], which inflates (deflates) the covariance (information) matrix. An interesting effect can be seen when comparing robots 1 and 3 to robot 5 - for the first two robots, the HS-CF is better, with a smaller RMSE and variance compared to the HS-CI, whereas on the other hand the results are opposite for robot 5, with the HS-CI giving a slightly better estimates. This is a surprising result, since recall that the homogeneous CI is an upper bound on all possible fusion covariances when considering all possible dependencies between two estimates (see [7]) – the HS-CI then should be an upper bound on the HS-CF results. One possible conclusion is that the HS-CF removes less ‘common data’ than it should. As this is beyond the scope of this paper, we leave this point to future research.

Robustness - Message Dropouts: The second set of simulations test the FG-DDF to message dropout scenarios, shown in Fig. 1(b). Here we again simulate the same 5-robot, 6-target scenario, only now each robot has only 90% (blue lines) or 50% (yellow lines) probability of actually receiving messages. Robots do not know whether the messages they sent are received by their neighbors. Note that the HS-CI fusion rule is relatively indifferent to this fact, as dependency information is not explicitly tracked. On the other hand, with the HS-CF, a dropped message means that the channel filters at both robots now hold a different estimate of the ‘common data’: the sending robot CF accounts for the data sent as common, while the other robot’s CF did not receive it. Fig. 1(b) compares the message dropout cases to a perfect communication scenario, and shows that despite the message dropout the FG-DDF performs well. The most significant influence is in the middle graph, showing robot 3, where we see a slight increase in the $2\sigma$ values for 90% dropout, and a more significant change, however still small (about 1.5 units at $t = 50s$) for the HS-CF. We can also see that as expected, in this scenario, message dropout has a larger effect when using the HS-CF fusion rule.

B. Hardware Experiment

To evaluate the robustness of the FG-DDF framework for message dropouts and measurement outliers, we deploy it on
two Clearpath Jackal UGV, shown in Fig. 2(c). The inference task of each robot is to estimate the 2D position and velocity of 3 of 5 targets \( t^m_k = [x^m_k, y^m_k, \dot{x}^m_k, \dot{y}^m_k]^T \) \((m = 1, 2, \ldots, 5)\), and its own constant (but unknown) robot-to-target relative position measurement bias \( b^i = [b^i_x, b^i_y]^T \), similar to [18].

The Jackals are equipped with a 2-core Intel Celeron G1840 CPU with 4GB of RAM and 128GB of disk drive storage and a 2-core Intel i7-7500U CPU with 32GB of RAM and 512GB of disk drive storage, respectively. They run the FG-DDF onboard as the inference and fusion engines, where ROS (version 1) is used for message passing. We use 5 Adeept wheeled robots for Arduino (AWR-A) as targets (see Fig. 2(c) and accompanying video). The targets are programmed to move in a straight line for about 4 seconds and then turn right for half a second.

We test the FG-DDF framework assuming linear measurement and dynamic models, despite that in practice the target motion is highly non-linear due to varied slipping characteristics. As in many target tracking problems, the target dynamics are modeled as ‘nearly constant velocity’ motion model [17]. The linear relative target and landmark position measurements are gathered using Vicon motion-capture cameras, corrupted by zero mean Gaussian noise, and are modeled as,

\[
\begin{align*}
    y^i_k &= t^m_k + s^i + v^i_{k,1}, \quad v^i_{k,1} \sim \mathcal{N}(0, R^i), \\
    m^i_k &= s^i + v^i_{k,2}, \quad v^i_{k,2} \sim \mathcal{N}(0, R^i).
\end{align*}
\]

Here the zero mean Gaussian noise for robots 1 and 2 is characterised by the covariance matrices \( R^1 = diag([1, 10]) \) and \( R^2 = diag([3, 3]) \), respectively.

Experiments results are shown in Fig. 2(a)-(b). In all experiments, robot 1 estimates its own bias states and targets 1-3 position and velocity state, similarly, robot 2 estimates it bias and targets 3-5, the robots then has one target in common (target 3). We performed 6 experiments using the HS-CI and HS-CF fusion rules with different communication success probabilities, and compare RMSE and 2\(\sigma\) bounds across each robot’s 14 states, based on truth values from the Vicon system. The two robots perform well, where

the error spike are contributed to outliers from the Vicon measurements, which occur when targets and robots pass too close to each other. The 2\(\sigma\) lines show similar behaviour to the one observed in simulations, where: (i) the HS-CF yields a more confident estimate than the HS-CI; (ii) for the HS-CF the 90\% communication success rate is almost identical to perfect communication, while for 50\% we see, as expected, a worse estimate; (iii) message dropouts have indistinguishable effects on the estimates in this scenario, which we attribute to the different measurement noise covariances between the robots, as this should cause the weight \( \omega \) (5) to approach 0, i.e. almost ignoring robot 1’s estimate.

V. CONCLUSIONS

As the size of robot teams and the variety of tasks they must perform increases, heterogeneous fusion becomes a core problem that must be addressed to ensure correct and scalable multi-robot information sharing for collaboration. In this paper we empirically test two heterogeneous fusion rules within the FG-DDF framework, namely the HS-CF and HS-CI, in realistic scenarios involving non-linear dynamic and measurement models, message dropouts, and measurement outliers. While under these conditions there are no formal guarantees for fusion to work (since common data dependencies are ill-defined), we show that consistent estimates can nevertheless still be produced for a challenging problems like multi-target tracking with self-localization.

Heterogeneous fusion allows robots to share only ‘relevant’ parts of their local pdfs, dramatically reducing communication and computation requirements for multi-robot teams. The empirical results presented and analyzed in this paper demonstrate that the FG-DDF is a practical framework for heterogeneous fusion under realistic conditions, such as in the presence of non-linearities and under significant message dropouts. From theoretical point of view, simulation results suggest that the HS-CF might not always remove common data dependencies correctly for all robots in non-linear problems, and leaves an open point for future research.
REFERENCES

[1] C. Y. Chong, E. Tse, and S. Mori, “Distributed estimation in networks,” in 1983 American Control Conference (ACC), Jun. 1983, pp. 294–300.

[2] O. Dagan and N. R. Ahmed, “Exact and approximate heterogeneous Bayesian decentralized data fusion,” May 2022, arXiv:2101.11116 [cs, eess] type: article. [Online]. Available: http://arxiv.org/abs/2101.11116v3

[3] A. Cunningham, V. Indelman, and F. Dellaert, “DDF-SAM 2.0: consistent distributed smoothing and mapping,” in 2013 IEEE International Conference on Robotics and Automation (ICRA), May 2013, pp. 5220–5227, iSSN: 1050-4729.

[4] M. Paskin and C. E. Guestrin, “Robust probabilistic inference in distributed systems,” in UAI 2004, 2004, arXiv: 1207.4174. [Online]. Available: http://arxiv.org/abs/1207.4174

[5] O. Dagan and N. R. Ahmed, “Factor graphs for heterogeneous Bayesian decentralized data fusion,” in 2021 IEEE 24th International Conference on Information Fusion (FUSION), Nov. 2021, pp. 1–8.

[6] S. Grime and H. Durrant-Whyte, “Data fusion in decentralized sensor networks,” Control Engineering Practice, vol. 2, no. 5, pp. 849–863, Oct. 1994.

[7] S. J. Julier and J. K. Uhlmann, “A non-divergent estimation algorithm in the presence of unknown correlations,” in Proceedings of the 1997 American Control Conference (ACC), vol. 4, Jun. 1997, pp. 2369–2373 vol.4.

[8] B. Noack, J. Sijs, M. Reinhardt, and U. D. Hanebeck, “Decentralized data fusion with inverse covariance intersection,” Automatica, vol. 79, pp. 35–41, May 2017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0005109817300298

[9] O. Dagan and N. R. Ahmed, “Conservative filtering for heterogeneous decentralized data fusion in dynamic robotic systems,” arXiv:2203.07142 [cs], Mar. 2022, arXiv: 2203.07142. [Online]. Available: http://arxiv.org/abs/2203.07142

[10] A. Makarenko, A. Brooks, T. Kaupp, H. Durrant-Whyte, and F. Dellaert, “Decentralised data fusion: A graphical model approach,” in 2009 12th International Conference on Information Fusion, Jul. 2009, pp. 545–554.

[11] C.-Y. Chong and S. Mori, “Graphical models for nonlinear distributed estimation,” in Proceedings of the 7th International Conference on Information Fusion (FUSION), Stockholm, Sweden, 2004, pp. 614–621.

[12] T. W. Martin and K. C. Chang, “A distributed data fusion approach for mobile ad hoc networks,” in 2005 International Conference on Information Fusion (FUSION), vol. 2, Jul. 2005, pp. 1062–1069.

[13] B. J. Frey, F. R. Kschischang, H.-A. Loeliger, and N. Wiberg, “Factor graphs and algorithms,” in In Proc. 35th Allerton Conf. Communications, Control, and Computing, 1997, pp. 666–680.

[14] F. Dellaert, “Factor graphs: exploiting structure in robotics,” Annual Review of Control, Robotics, and Autonomous Systems, vol. 4, no. 1, pp. 141–166, 2021.

[15] R. Eastic, H. Singh, and J. Leonard, “Exactly Sparse Delayed-State Filters,” in Proceedings of the 2005 IEEE International Conference on Robotics and Automation. Barcelona, Spain: IEEE, 2005, pp. 2417–2424. [Online]. Available: http://ieeexplore.ieee.org/document/1576475

[16] X. Rong, L. Zhanlue, Z. Vesselin, and P. Jilkov, “Practical measures and test for credibility of an estimator,” in Proc. Workshop on Estimation, Tracking, and Fusion—A Tribute to Yaakov Bar-Shalom, 2001.

[17] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, “Linear estimation in static systems,” in Estimation with Applications to Tracking and Navigation. John Wiley & Sons, Ltd, 2001, pp. 121–177.

[18] B. Noack, S. J. Julier, and U. D. Hanebeck, “Treatment of biased and dependent sensor data in graph-based SLAM,” in 2015 18th International Conference on Information Fusion (Fusion), Jul. 2015, pp. 1862–1867.