MH-iSAM2: Multi-hypothesis iSAM using Bayes Tree and Hypo-tree

Ming Hsiao and Michael Kaess

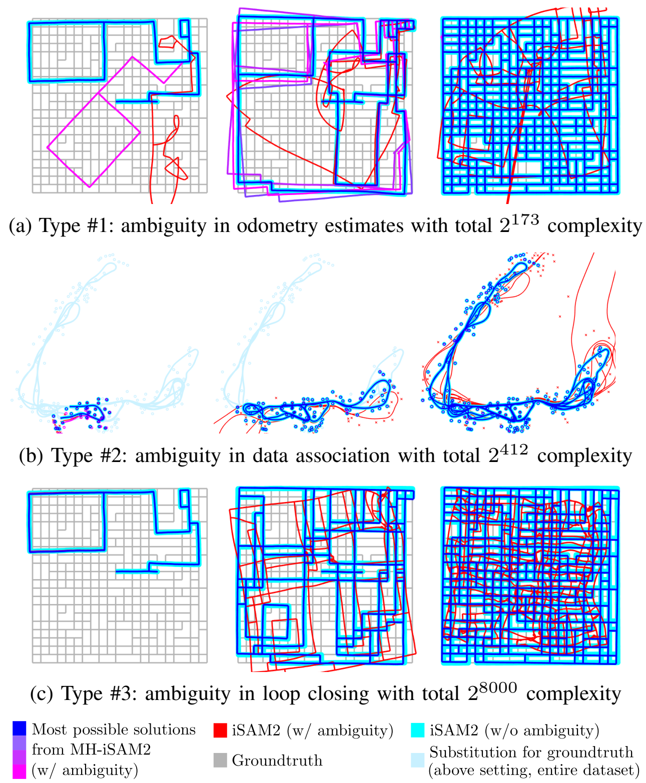
***Abstract*— A novel nonlinear incremental optimization algorithm MH-iSAM2 is developed to handle ambiguity in simultaneous localization and mapping (SLAM) problems in a multi-hypothesis fashion. It can output multiple possible solutions for each variable according to the ambiguous inputs, which is expected to greatly enhance the robustness of autonomous systems as a whole. The algorithm consists of two data structures: an extension of the original *Bayes tree* that allows efficient multi-hypothesis inference, and a *Hypo-tree* that is designed to explicitly track and associate the hypotheses of each variable as well as all the inference processes for optimization. With our proposed hypothesis pruning strategy, MH-iSAM2 enables fast optimization and avoids the exponential growth of hypotheses. We evaluate MH-iSAM2 using both simulated datasets and real-world experiments, demonstrating its improvements on the robustness and accuracy of SLAM systems.**

I. Introduction

The robustness of simultaneous localization and mapping (SLAM) is crucial to mobile robots. However, it is hard to achieve under common SLAM frameworks, which assume that the *back-end* (optimizer) always get correct and unbiased information from the *front-end* (data processing), and outputs only one solution for each unknown variable. As a result, when ambiguities occur (e.g. a feature point is detected to be very similar to more than one landmark, or two loop closure candidates are found but contradict one another), the front-end might not be able to determine which information is correct, and wrong information can be added into the back-end optimization, which can pollute the SLAM system and lead to the failure of the entire robotic system.

It would be desirable for the back-end solver to explicitly account for the ambiguities that cannot be handled by the front-end, and to output all the highly probable solutions. This proposed framework allows the later modules in the robotic system (e.g.: control or planning) to be aware of the temporarily unsolvable ambiguities and therefore is expected to greatly enhance the robustness of the entire robotic system.

Based on the *incremental smoothing and mapping using Bayes tree (iSAM2)* [11] algorithm, we develop a novel online nonlinear incremental optimizer called MH-iSAM2, which takes multi-mode measurements that model the ambiguities as inputs, and generates multi-hypothesis outputs which are the exact solutions of the most possible results. There are three main parts of the MH-iSAM2 algorithm. The first part is a *multi-hypothesis Bayes tree (MHBT)* that allows efficient inference among the *multi-mode factors (MMF)* and *multi-hypothesis variables (MHV)*, which is modified from the original Bayes tree [11]. The second part is a novel data structure called *Hypo-tree*, which tracks the modes of factors together with the hypotheses of variables and other components in the inference process in the MHBT, and associates them for computations in the optimization. The third part is a pruning algorithm that selects the unwanted hypotheses and prune them in both Hypo-tree and MHBT.



(a) Type #1: ambiguity in odometry estimates with total 2^{173} complexity

(b) Type #2: ambiguity in data association with total 2^{412} complexity

(c) Type #3: ambiguity in loop closing with total 2^{8000} complexity

Fig. 1: Examples of MH-iSAM2 tracking and solving various types of ambiguities (see Sec. III-C) in SLAM problems online incrementally as more observations are added into the system (left to right). Multiple most possible solutions can be solved from (a) a pose graph with ambiguous odometry measurements, (b) a SLAM problem with ambiguous data association of 2D feature points, and (c) a pose graph with ambiguous loop closures.

The contributions of this work are:

1. Developing and open-sourcing the novel multi-hypothesis nonlinear incremental SLAM solver MH-iSAM2 (available at: https://bitbucket.org/rpl_cmu/mh-isam2_lib),
2. Modeling different types of ambiguities in SLAM problems in a multi-hypothesis factor graph,
3. Proposing the idea of local hypotheses and using the Hypo-tree as a novel data structure to track all of them,
4. Extending Bayes tree and its inference algorithm to solve the mutli-hypothesis SLAM problems efficiently,
5. Designing a pruning algorithm that removes the unwanted hypotheses to maintain efficiency, and
6. Evaluating MH-iSAM2 using simulated datasets as well as demonstrating a real-world application for it.

The authors are with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, USA. {mhsiao, kaess}@cmu.edu

This work was partially supported by Lockheed Martin Corporation under research agreement A020478. We would like to thank C. Debrunner, S. Tangirala, K. Anderson, and D. Fanning for discussions and feedback, and E. Westman and Y. Yang for discussions of the implementation.

II. Related Work

Many previous studies focus on solving ambiguity problems either in the front-end, such as RANSAC [6] and JCBB [16], or of a specific type only, such as loop closing [13][2][21][3]. However, none of them can deal with general ambiguities that are unsolvable by the front-end.

FastSLAM [14] can handle unknown data association though sampling particles over all possibilities, and [15][7] apply nonparametric methods to deal with ambiguities in data association or loop closing. These approaches in theory model all possibilities, but can only approximate the best solution(s) instead of solving for the exact values. [9] tracks multiple hypotheses of exact solutions as desired. However, it requires additional batch steps for relinearization, and can be improved for SLAM applications, such as tracking the hypothesis of each variable more efficiently (see Sec. IV-A).

There are other back-end solutions that change the topology of the underlying graph [19][20] or self-tuning the distribution parameters [17] during optimization with ambiguity, or add a step before conventional optimizers to choose the best hypothesis [18]. However, none of them can model more than one mode in the final output.

We develop MH-iSAM2 based on iSAM2 [11] instead of other open source solvers [10][12][1] because it allows online nonlinear incremental updates using Bayes tree for efficiency. Most of the advantages of iSAM2 are still preserved in MH-iSAM2, which is further discussed in Sec. V.

III. Multi-hypotheses Models of Ambiguity

A. From Probability to Hypotheses

From the probabilistic point of view, SLAM can be modeled as a maximum likelihood estimation (MLE) problem:

$$\hat{\Theta} = \arg\max_{\Theta} P\left(Z|\Theta\right) = \arg\max_{\Theta} \prod_{k} P\left(z_k|\Theta_k\right), \quad (1)$$

where Z is the set of all measurements z_k that are independent to each other. Θ is the set of all variables θ_l, while $\Theta_k \subseteq \Theta$ is the subset of variables that directly affect z_k in the conditional probability $P\left(z_k|\Theta_k\right)$. $\hat{\Theta}$ is the set of solutions of all variables that maximize $P\left(Z|\Theta\right)$.

Most existing SLAM back-ends assume that every measurement z_k is sampled from an single Gaussian distribution $\mathcal{N}(\mu_k, \Sigma_k)$ with mean μ_k and covariance Σ_k, and the resulting distribution of $P\left(Z|\Theta\right)$ is also a single Gaussian $\mathcal{N}(\mu_{\mathrm{L}}, \Sigma_{\mathrm{L}})$. However, this assumption no longer holds for ambiguous measurements because a single Gaussian distribution only has one peak and cannot model more than one mode. A simple extension is to use a *Gaussian mixture model (GMM)* to represent an ambiguous measurement z_r with multiple modes $z_{r(i)}$, which can be written as

$$P_{\mathrm{M}}(z_r|\Theta_r) = \sum_{i=0}^{m_r} w_{r(i)} \mathcal{N}\left(\mu_{r(i)}, \Sigma_{r(i)}\right) = \sum_{i=0}^{m_r} w_{r(i)} P\left(z_{r(i)}|\Theta_r\right) \quad (2)$$

where m_r is the number of modes in z_r, and $w_{r(i)}$ is a weighting for each $\mathcal{N}\left(\mu_{r(i)}, \Sigma_{r(i)}\right)$ which satisfies $\sum_{i=0}^{m_r} w_{r(i)} = 1$. The subscript "$_{(i)}$" indicates each mode i of z_r. Assuming that every z_r is independent to all others, we can rewrite the MLE problem in Eq. 1 as

$$\hat{\Theta} = \arg\max_{\Theta} \left[\prod_{s} P\left(z_s|\Theta_s\right)\right]\left[\prod_{r} P_{\mathrm{M}}\left(z_r|\Theta_r\right)\right] \quad (3)$$

to represent SLAM problems with ambiguity, where each index s corresponds to a single-mode measurement z_s while index r corresponds to a multi-mode measurement z_r. Even though we can still solve this MLE problem and get a single estimation $\hat{\Theta}$ that corresponds to one of the highest peaks of the resulting GMM, the information of all other peaks that result from other combinations of modes is lost.

To preserve all the combinations of modes, we represent the problem differently as a *multi-hypothesis MLE problem (MH-MLE)*:

$$\hat{\mathbf{\Theta}}_{\mathrm{M}} = \left\{\hat{\Theta}_{[\mathbf{i}]} | \mathbf{i} \in \mathbb{N}^t\right\}, \quad (4)$$

$$\hat{\Theta}_{[\mathbf{i}]} = \arg\max_{\Theta} \left[\prod_{s} P(z_s|\Theta_s)\right]\left[\prod_{r=1}^{t} w_{r(i_r)} P\left(z_{r(i_r)}|\Theta_r\right)\right], \quad (5)$$

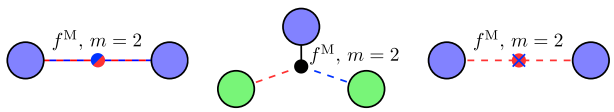
where t is the total number of multi-mode measurements z_r. Index $\mathbf{i}$ is a t dimensional vector whose r-th element i_r indicates the choice of mode of z_r, and the entire $\mathbf{i}$ vector represents one *overall hypothesis* $h_j^{\{t\}}$ (j is a scalar index that 1-to-1 associates with $\mathbf{i}$), which is one of the combinations of all modes (we use "mode" for inputs and "hypothesis" for outputs in this paper). Since only each $\hat{\Theta}_{[\mathbf{i}]}$ instead of the joint distribution is of interest, we can regard Eq. 5 as a MLE problem and solve each of them individually. In other words, a MH-MLE problem is actually a set of MLE problems with each corresponds to one $h_j^{\{t\}}$ and one $\hat{\Theta}_{[\mathbf{i}]}$. So, the entire set $\hat{\mathbf{\Theta}}_{\mathrm{M}}$ covers all the combinations of all modes of all z_r.

B. Multi-hypothesis Factor Graph (MHFG)

An MH-MLE problem can be represented in a *multi-hypothesis factor graph (MHFG)*, which is an extension of the original factor graph [5] and can be converted into a multi-hypothesis Bayes tree (MHBT) and solved efficiently (see Sec. V). An MHFG consists of *single-mode factors (SMF), multi-mode factors (MMF)*, and *Multi-hypothesis variables (MHV)*. An SMF corresponds to one $P\left(z_s|\Theta_s\right)$ in Eq. 5, which is the same as a factor in the original factor graph. An MMF models each mode of an ambiguous measurement as a individual Gaussian distribution as described in Eq. 5. Three types of MMFs are defined in Sec. III-C). An MHV $\theta_p \in \hat{\mathbf{\Theta}}_{\mathrm{M}}$ can represent its multiple values from each hypothesis in an efficient way (see Sec. III-D and IV-A).

C. Multi-mode Factors (MMF)

We define three types of multi-mode factors (MMF) f^{M}, each with m modes ($m > 1$), to model most kinds of discrete ambiguities that cannot be solved by front-ends. Type #1 is the *multi-measurement factor*, which consists of m various measurements that are all connected among the same MHVs (see Fig. 2-a). For example, two different visual odometry (VO) estimates can be loosely-coupled in one pose graph for better accuracy. However, when the two estimates are very different, it is very likely that one of them is an outlier and



(a) Multi-measurement (b) Multi-association (c) Boolean

Fig. 2: The three types of MMFs f^{M}. Red and blue show the two modes ($m=2$). Purple and green nodes are poses and landmarks respectively.

should not be considered when computing the optimal result. In this case, we can model them as a type #1 MMF with two modes.

Type #2 is the *multi-association factor*, which contains only one measurement but is connected among m MHVs that are the same type and at least one other MHV (see Fig. 2-b). For example, when a newly observed feature point is very similar to more than one landmark both geometrically and in appearance, the front-end again cannot tell which is the accurate association without any other information.

Type #3 is the *Boolean factor*, which represents whether a factor should exist or not ($m=2$). One common application is to model each loop closure candidate in a loop closing ambiguity problem [13][2][21] (see Fig. 2-c).

D. Multi-hypothesis Variables (MHV)

A multi-hypothesis variable (MHV) θ_p contains multiple estimates for one variable, each corresponds to one hypothesis of θ_p. The hypotheses of θ_p are determined by all the MMFs that affect it, which is hard to be tracked since it depends on the topological structure of the MHFG. Therefore, we introduce the Hypo-tree data structure in Sec. IV to simplify the hypothesis tracking process.

Another challenge is that there can be causal relationship among MMFs, e.g.: the previous choice of closing a loop or not can affect a later data association, so the hypotheses of the affected θ_p are even harder to track. However, we can still assume that all of the MMFs are independent to each other as defined in MH-MLE without losing generality. Even though each θ_p might contain redundant hypotheses from impossible combinations of modes, it at least preserves all the possible hypotheses, and those redundancies can be removed later through hypotheses pruning (see Sec. VI).

IV. Hypotheses Tracking in Hypo-tree

A. Overall and Local Hypotheses

Because of the independence assumption (see Sec. III), when multiple MMFs exists in one MHFG, the number n_t of the *overall hypotheses* $h^{\{t\}}=\left\{h_j^{\{t\}}|0\leq j<n_t\right\}$ of the entire system is $n_t=\prod_{r=1}^{t} m_r$, where m_r is the number of modes of each MMF f_r^{M}, and t is the total number of MMFs. Even though n_t grows exponentially to t and has to be pruned (see Sec. VI) to maintain a tractable size, the number of *local hypotheses* $h^{\{r\}}=\left\{h_j^{\{r\}}|0\leq j<n_r\right\}$ of each MHV θ_p can be less than n_t (see Fig. 3). As a result, we track $h^{\{r\}}$ of each θ_p instead of $h^{\{t\}}$ to improve efficiency.

However, $h^{\{r\}}$ of each θ_p can change as more measurements (SMFs or MMFs) are added into the system. For example, as shown in Fig. 4-a, if a loop closure is added into

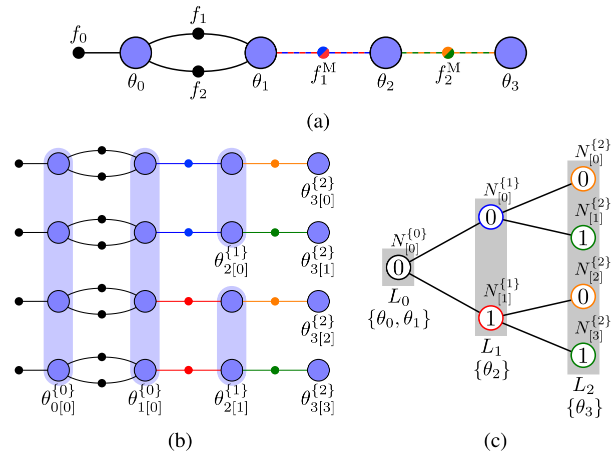


Fig. 3: An example of hypotheses growing. (a) A MHFG with two MMFs (each with 2 modes: blue/red and orange/green) can be regarded as 4 individual factor graphs in (b). However, the value of some MHVs (θ_0, θ_1, and θ_2) are the same in different factor graphs (purple shadows), which implies that only fewer hypotheses are needed to model these θ_p. (c) We can associate each θ_p with a Hypo-layer L_r in Hypo–tree to track its hypotheses (written as $\{\cdot\}$). The number in each Hypo-node indicates its mode.

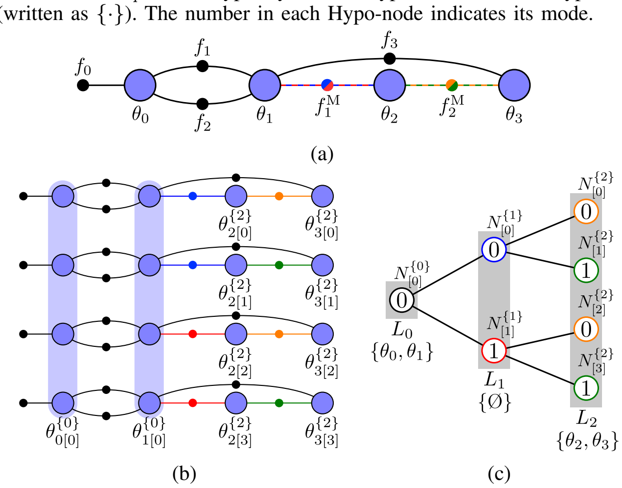
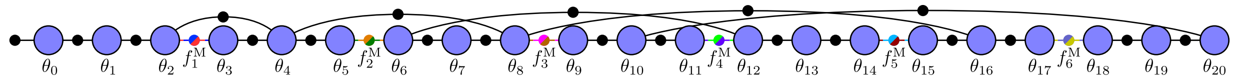


Fig. 4: An example of hypotheses expansion. (a) Factors that link the current variable with an earlier one can result in hypotheses expansion of MHV, which can be observed by comparing θ_2 in (b) and Fig. 3-b. (c) In this case, we only have to update the association between Hypo-layers and MHVs (move θ_2 from L_1 to L_2) without changing the structure of Hypo-tree.

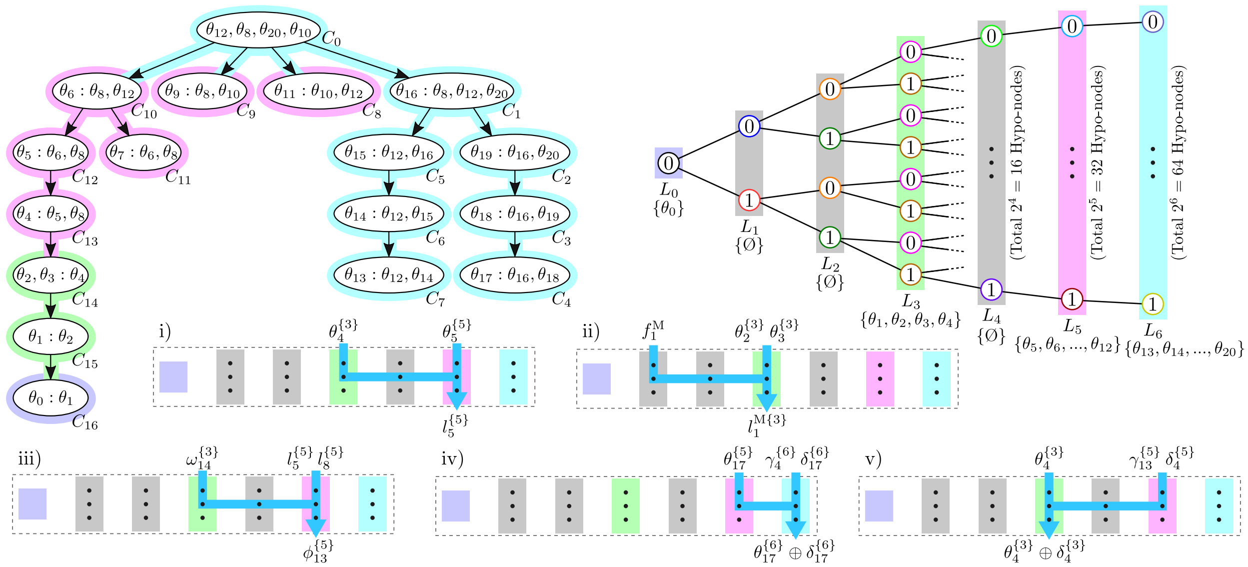
the MHFG in Fig. 3-a, some of the θ_p begin to be affected by the MMFs that originally do not affect θ_p (see Fig. 4-b). Therefore, those θ_p have to expand their $h^{\{r\}}$ accordingly.

B. Construction of Hypo-tree

To handle both the growing of $h^{\{t\}}$ and the expansion of $h^{\{r\}}$ efficiently, we propose the Hypo-tree data structure (see Fig. 3-c and 4-c). It consists of several *Hypo-layers* L_r, each results from one MMF f_r^{M} following the temporal ordering $r=0,...,t$ and contains several *Hypo-nodes* $N_{[j]}^{\{r\}}$ $(j=0,...,n_r)$ that represent local hypotheses $h_{[j]}^{\{r\}}$. Starting from L_0 that contains only one Hypo-node $N_{[0]}^{\{0\}}$, whenever a new MMF f_{t+1}^{M} is observed, a new Hypo-layer L_{t+1} will be created, and m_{t+1} new Hypo-nodes $N_{[j']}^{\{t+1\}}$ will be generated in L_{t+1} as the children of each $N_{[j]}^{\{t\}}$ in L_t. Therefore, the total number of Hypo-nodes in L_{t+1} is $n_{t+1}=n_t{\cdot}m_{t+1}$. Due to this incremental construction procedure, the topologies of previous layers $L_0,\ldots,L_t$ never change.



(a) A MHFG with 21 poses, 20 odometry factors (14 SMFs and 6 type #1 MMFs), and 5 loop closures



(b) The associations between the MHBT (left) and the Hypo-tree (right), and finding the correspondences of local hypotheses

Fig. 5: An example of inference process in a MHBT with the help of Hypo-tree. (a) A SLAM problem with ambiguity is represented as a MHFG. (b) The corresponding MHBT and Hypo-tree are constructed from the MHFG and associated with each other (the MHCD $\gamma_q^{\{r\}}$ in each clique C_q is associated with the Hypo-layer L_r that is colored the same as the shadow of C_q). Hypo-tree is used to not only find the correspondences among the modes of MMFs f_r^{M} and the hypotheses of MHVs $\theta_p^{\{r\}}$, MHJDs $\phi_q^{\{r\}}$, MHCDs $\gamma_q^{\{r\}}$, and MHMDs $\omega_q^{\{r\}}$, but also determine the output hypotheses of $\theta_p^{\{r\}}$, $\phi_q^{\{r\}}$, and $\gamma_q^{\{r\}}$. For example: i) Linearization of the SMF between θ_4 and θ_5. ii) Linearization of the MMF f_1^{M}. iii) Forming the MHJDs $\phi_{13}^{\{r\}}$ of clique C_{13}. iv) Retraction (denoted as $\oplus$) of θ_{17} in C_4, assuming the loop closing factor between θ_{10} and θ_{20} is newly added in this iteration (expanding the hypotheses of θ_{17}). v) Retraction of θ_4 in C_{13} (merging the hypotheses of δ_4).

C. Association and Correspondence

Every new MHV θ_p is associated with the latest Hypo-layer L_r once added into the system, which can be denoted as $\theta_p^{\{r\}}$. Also, each value of $\theta_p^{\{r\}}$ (denoted as $\theta_{p[j]}^{\{r\}}$) is associated with each Hypo-node $N_{[j]}^{\{r\}}$ (or local hypothesis $h_{[j]}^{\{r\}}$) in L_r, for $0 \leq j < n_r$. When the local hypotheses of θ_p have to be expanded (e.g. the example in Sec. IV-A), we only have to update the association of $\theta_p^{\{r\}}$ to a different Hypo-layer $L_{r'}$, which can be denoted as $\theta_p^{\{r\}} \rightarrow \theta_p^{\{r'\}}$, and expand the number of values in $\theta_p^{\{r'\}}$ accordingly.

Based on the association, searching for the corresponding values of local hypotheses between variables is simple. The $h_{[j^*]}^{\{r^*\}}$ of a MHV $\theta_{p^*}^{\{r^*\}}$ that is the ancestor of $h_{[j]}^{\{r\}}$ of another $\theta_p^{\{r\}}$ can be found through traversing from $N_{[j]}^{\{r\}}$ towards the root till reaching L_{r^*} ($r^* < r$). Or, the set of local hypotheses $\left\{h_{[j']}^{\{r'\}} | 0 \leq j' < n_{r'}\right\}$ of $\theta_{p'}^{\{r'\}}$ that are the descendants of $h_{[j]}^{\{r\}}$ of $\theta_p^{\{r\}}$ can be found through traversing from $N_{[j]}^{\{r\}}$ towards the leaves till reaching $L_{r'}$ ($r' > r$). Same method is applied to search for the hypotheses correspondences among other components in the inference process (see Sec. V).

V. Inference in Multi-hypothesis Bayes Tree

A. Multi-hypothesis Bayes Tree (MHBT)

Multi-hypothesis Bayes tree (MHBT) is an extension of the original Bayes tree [11] that conducts efficient inference for a MHFG. A MHBT stores *multi-hypothesis conditional densities (MHCD)* $\gamma_q^{\{r\}}$ in each of its cliques C_q, and applies a multi-hypothesis inference process throughout all the cliques (see Fig. 5) to solve for all the MHVs.

A MHBT can be constructed from a MHFG based on an ordering of MHVs. In each clique C_q, the relevant SMFs, MMFs, and all of the *multi-hypothesis marginal densities (MHMD)* $\omega_{q^*}^{\{r^*\}}$ that are passed from the children cliques C_{q^*} (if any) are combined into a *multi-hypothesis joint density (MHJD)* $\phi_q^{\{r\}}$. Then, $\phi_q^{\{r\}}$ is factorized into a MHCD $\gamma_q^{\{r\}}$ and a MHMD $\omega_q^{\{r\}}$. Finally, $\omega_q^{\{r\}}$ are passed to the parent clique $C_{q'}$. Repeating this process from leaves to root completes one iteration of inference, and several iterations are needed before convergence for a nonlinear SLAM problem. Notice that each $\phi_q^{\{r\}}$, $\gamma_q^{\{r\}}$, and $\omega_q^{\{r\}}$ is associated with a Hypo-layer L_r for the search of hypotheses correspondences among them (see Sec. IV-C).

B. Incremental Update and Reordering

Because of the causality property of the growth of hypotheses, constructing MHBT incrementally from a growing MHFG can achieve better efficiency than batch. As new factors being added into the corresponding MHFG, the top part of the MHBT can be rebuilt without changing the subtrees that are not directly linked with the new factors. Also, the hypotheses of the MHVs in those subtrees can stay changed except for the hypotheses expansion of MHVs during backsubstitution. In our implementation, the updated MHVs in each iteration are reordered based on the CCO-

LAMD algorithm [4] (also applied in [11]). Moreover, since all hypotheses have to share the same ordering, all the edges that represent a mode in each multi-association factor are regarded as connected, and all Boolean factors are regarded as true (connected) in the ordering process.

C. Linearization

Each nonlinear factor f_s (a SMF) or f_r^{M} (a MMF) is linearized with respect to a linearization point of all the relevant MHVs Θ_s or Θ_r (see Sec. III-A) if required (*fluid relinearization* [11] is applied). Consequently, each local hypothesis $h_{[j]}^{\{r'\}}$ of a *multi-hypothesis linearized factor (MHLF)* $l_s^{\{r'\}}$ (from f_s) or $l_r^{\mathrm{M}\{r'\}}$ (from f_r^{M}) is calculated by finding the correspondences among the local hypotheses of Θ_s or Θ_r, and also the corresponding modes of f_r^{M} for $l_r^{\mathrm{M}\{r'\}}$ only (see Fig. 5-b). Notice the associated Hypo-layer $L_{r'}$ of $l_r^{\mathrm{M}\{r'\}}$ might not be the same as L_r that results from f_r^{M} since some of its relevant MHVs θ_p might be affected by later MMFs (thus $r' \geq r$). Based on the Gaussian assumption in Eq. 5, each MHLF is a set of Jacobian matrices $\mathbf{A}^{\{r'\}} = \left\{A_{[j]}^{\{r'\}} | 0 \leq j < n_{r'}\right\}$, and each $A_{[j]}^{\{r'\}}$ contains the right-hand-side (RHS) vector as an additional column in practice. Then, their Hessian matrices $\Lambda_{[j]}^{\{r'\}} = \left(A_{[j]}^{\{r'\}}\right)^{\top}\left(A_{[j]}^{\{r'\}}\right)$ that represents the local densities are generated for the next step.

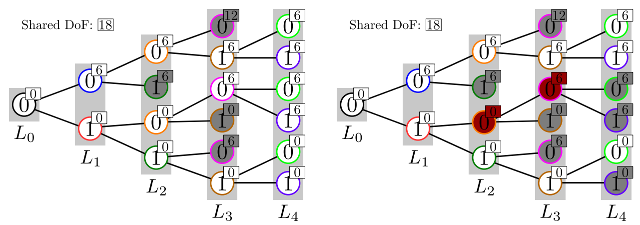
D. Clique-based Elimination

Based on the ordering of MHVs in Sec. V-B and the clique formation algorithm in [11], each MHLF is grouped into one of the cliques. Then, the MHJD $\phi_q^{\{r'\}}$ of each clique C_q can be generated as a set of Hessian matrices $\mathbf{\Phi}_q^{\{r'\}} = \left\{\Phi_{q[j]}^{\{r'\}} | 0 \leq j < n_{r'}\right\}$ by summing up the corresponding $\Lambda_{[j]}^{\{r\}}$ and input MHMDs $\omega_{q^*}^{\{r^*\}}$ from its children C_{q^*}, which is also a set of Hessian matrices . Because of the incremental update strategy (see Sec. V-B), the number of hypotheses $n_{r'}$ of $\phi_q^{\{r'\}}$ in clique C_q is no less than n_{r^*} of any of its input MHMDs $\omega_{q^*}^{\{r^*\}}$ ($r' \geq r^* \Rightarrow n_{r'} > n_{r^*}$). As a result, one matrix of an input $\omega_{q^*}^{\{r^*\}}$ can be reused by more than one $\Phi_{q[j]}^{\{r'\}}$ of $\phi_q^{\{r'\}}$, which is more efficient than conducting the same process in individual Bayes trees for each $h_{[j]}^{\{t\}}$.

Finally, we apply *partial Cholesky factorization* on each $\Phi_{q[j]}^{\{r'\}}$ of $\mathbf{\Phi}_q^{\{r'\}}$ based on the *frontals* Θ_q^{F} and *separators* Θ_q^{S} in C_q (as defined in [11]) and *eliminate* the frontals Θ_q^{F} from the rest of the inference process. The outputs are a MHCD $\gamma_q^{\{r'\}}$ in the form of a set of factorized Jacobian matrices and a MHMD $\omega_q^{\{r'\}}$ again in the form of a set of Hessian matrices. In practice, we cache $\omega_q^{\{r'\}}$ as in [11] to save computations in incremental updates (see Sec. V-B).

E. Backsubstitution and Retraction

Following the backsubstitution algorithm in [11], we can solve the *multi-hypothesis linear updates (MHLU)* $\delta_p = \left\{\delta_{p[j]} | 0 \leq j < n_{r''}\right\}$ of each frontal $\theta_p^{\mathrm{F}} \in \Theta_0^{\mathrm{F}}$ from the root to all the leaves. However, since δ_p is calculated from $\gamma_q^{\{r\}}$ and each $\delta_{p'}$ of its corresponding separator $\theta_{p'} \in \Theta_q^{\mathrm{S}}$, the number $n_{r''}$ of hypotheses of δ_p is determined by the largest number of hypotheses among $\gamma_q^{\{r\}}$ and all $\delta_{p'}$, which might be greater than n_r of $\theta_p^{\{r\}}$ and even contain redundant duplicated values. Thus, we first try to merge numerically similar values in δ_p based on their hypotheses correspondences. Then, if the number $n_{r'''}$ of hypotheses of δ_p is still larger than that of $\theta_p^{\{r\}}$ after merging, we expand the hypotheses of $\theta_p^{\{r\}}$ to match with it ($\theta_p^{\{r\}} \rightarrow \theta_p^{\{r'''\}}$, see Fig. 5-b) for retraction.



(a) Before pruning in L_4 (b) After pruning in L_4

Fig. 6: An example of hypotheses pruning and DoF recording in Hypo-tree, where f_1^{M} and f_3^{M} (associated to L_1 and L_3) are type #3 MMFs. With pruning in L_1~L_3 done (dark gray nodes) in previous updates (a), we apply pruning in L_4 with backward pruning (dark red nodes) (b). The part of DoF that results from type #3 MMFs (which can be different for each $h_{[j]}^{\{t\}}$) is stored in each $h_{[j]}^{\{t\}}$ (small boxes) while the rest part of DoF that results from all other factors is shared by all $h_{[j]}^{\{t\}}$. E.g.: the DoF of $h_{[1]}^{\{4\}}$ is $18+6=24$.

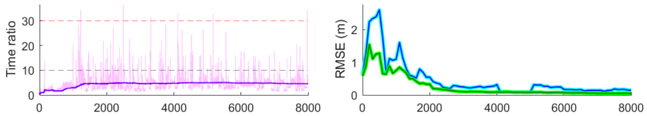
VI. Hypotheses Pruning

A. Pruning Criteria

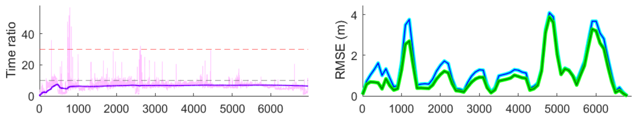
The unwanted and unlikely hypotheses (see Sec. III-D and IV-A) are pruned to maintain efficiency right after the elimination step (see Sec. V-D). First, every overall hypothesis $h_{[j]}^{\{t\}}$ in the the latest Hypo-layer L_t with its corresponding *squared system error* $e_{[j]}^2$ larger than its 95% chi-square threshold $\chi_{[j]}^2$ is pruned. If the number n_{R_1} of remaining $h_{[j]}^{\{t\}}$ is greater than a threshold n_{desire}, we further prune those $h_{[j]}^{\{t\}}$ with fewer degrees of freedom (DoF) $d_{[j]}^{\{t\}}$ (defined as "dimension of all factors"−"dimension of all variables", which is also used to calculate $\chi_{[j]}^2$). Then, if the number n_{R_2} of remaining $h_{[j]}^{\{t\}}$ is greater than another threshold n_{limit} (e.g. all $d_{[j]}^{\{t\}}$ are the same), we prune those with lower chi-square probabilities one-by-one until the number n_{R_3} of remaining $h_{[j]}^{\{t\}}$ is smaller than n_{limit} ($n_{\mathrm{limit}} \geq n_{\mathrm{desire}}$ allows tracking more $h_{[j]}^{\{t\}}$ when too many of them all seem likely). In practice we use the error $e'_{[j]}$ of the linearized system in each iteration to approximate $e_{[j]}$, which is calculated in the elimination step (see Sec. V-D) as the bottom right element of the matrix $\Gamma_{0[j]}^{\{t\}}$ of the MHCD $\gamma_0^{\{t\}}$ of the root clique C_0. Each DoF $d_{[j]}^{\{t\}}$ is recorded in two parts as shown in Fig. 6.

B. Pruning in Both Trees

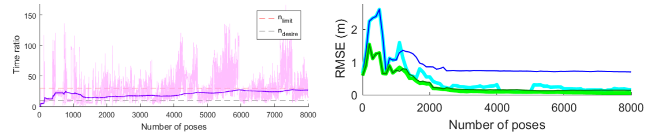
Once an overall hypothesis $h_{[j]}^{\{t\}}$ is pruned, we flag the corresponding Hypo-node $N_{[j]}^{\{t\}}$ as pruned, and no children will be created from $N_{[j]}^{\{t\}}$ (see Fig. 6-a). Also, after finishing flagging in the last Hypo-layer L_t, we check every unflagged



(a) Type #1: Multi-measurement factor (based on city10000 dataset)



(b) Type #2: Multi-association factor (based on Victoria Park dataset)



(c) Type #3: Boolean factor (based on city10000 dataset)

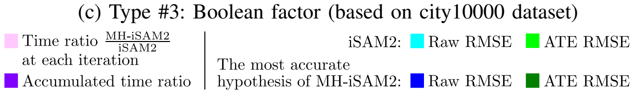
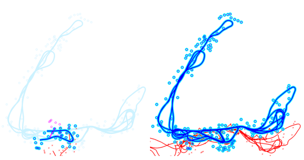
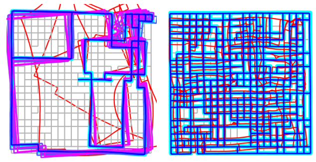


Fig. 7: Speed (left column) and accuracy (right column) analysis of each type of MMF. Notice that in (b) we take the complete iSAM2 result as groundtruth since the real groundtruth is unavailable.



(a) Type #1 and type #2



(b) Type #1 and type #3

Fig. 8: The example results of MH-iSAM2 with two types of ambiguities.

Hypo-node $N_{[j]}^{\{r\}}$ from L_{t-1} to L_1 if all its children are flagged (see Fig. 6-b). If so, we flag $N_{[j]}^{\{r\}}$ as well. Then, the associated values of those flagged $N_{[j]}^{\{r\}}$ in each MHV $\theta_p^{\{r\}}$ and MHCD $\gamma_q^{\{r\}}$ of the MHBT are removed immediately. Notice that we only remove the associated $\Omega_{q[j]}^{\{r\}}$ of a cached MHMD $\omega_q^{\{r\}}$ when it is used in the incremental update step.

VII. Experimental Results

A. Experimental Settings

We evaluate the accuracy and efficiency of MH-iSAM2 through simulations and a real-world experiment. The algorithm is implemented in C++ and executed on a desktop with an Intel Core i7-4790 processor. Whenever a new observation is added into the system, we run one iteration of update and calculate the most up-to-date estimates of all MHVs.

B. Simulation Results

All three types of ambiguities (see Sec. III-C) are simulated based on the *city10000* or *Victoria Park* dataset by randomly adding wrong measurements into them. The example outputs are shown in Fig. 1. From Fig. 7, we can tell that the speed of MH-iSAM2 is constant to the *overall complexity of ambiguity* (defined as $\prod_{r=1}^{t} m_r$). Moreover, MH-iSAM2 is efficient enough to track up to $n_{\text{limit}} = 30$ hypotheses within less than $30\times$ of time of the original iSAM2. However, the speed varies in each iteration due to the extra computation for hypothesis handling. Also, since each mode of every type #2 or #3 MMF affects the topology and density of MHFG, it affects the speed as well.

Method	RMSE
iSAM2 (o1)	0.140m
iSAM2 (o2)	0.359m
iSAM2 (joint)	0.132m
MH-iSAM2	**0.129m**

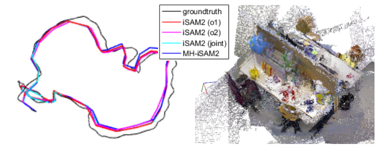


Fig. 9: The RMSE evaluations, trajectories, and output map of the hypothesis with smallest error from MH-iSAM2 of a real-world dataset (left to right).

As for the accuracy analysis of the hypothesis with the smallest root-mean-square error (RMSE) in Fig. 7, we can tell that MH-iSAM2 can keep track of the "correct hypothesis" with reasonable cost of time. Although some wrong loop closures might be added into all remaining hypotheses (see Fig. 1-c and 7-c), their pollution to the outputs are hardly visible since they all pass the chi-square threshold.

C. Real-world Experiment

A pose graph is constructed by two odometry estimated using two different settings of the *fast dense RGB-D odometry* in [8] and loop closures detected and registered as in [8]. Following the idea in Sec. III-C, each two VO estimates between the same poses are modeled as a type #1 MMF if their difference is too large (e.g. one gets bad estimate), or loosely-coupled as two SMFs otherwise. From the results in Fig. 9, we find that MH-iSAM2 (with 2^5 complexity) outperforms the conventional single-mode framework that takes either each VO individually or both of them as input.

D. Discussions

Even though each test case in the analysis only contains one type of MMF, and each tested MMF only contains 2 modes, there is no constraint on the number of modes of each MMF or combinations of their types in the current MH-iSAM2 framework (see Fig. 8). However, MH-iSAM2 can still be improved and extended in several other aspects. First, the type #3 MMF can actually be combined with type #1 or type #2 MMF to represent an additional possibility that all the existed modes are invalid. Second, since the merging of δ_p does not consider the entire numerical changes until convergence, some of the θ_p might end up containing more hypotheses than needed after being expanded (e.g.: θ_1 in Fig. 5). Third, n_{desire} and n_{limit} should be adjusted online based on the current complexity of ambiguity to avoid losing track of the correct hypotheses. Lastly, even though current MH-iSAM2 framework deals with discrete ambiguity only, modeling the degeneracy and continuous ambiguity in the same framework seems possible yet requires more studies.

VIII. Conclusion

We present the novel online incremental nonlinear optimizer MH-iSAM2 to handle the ambiguities in SLAM. Based on the Hypo-tree, MHBT, and the hypothesis pruning algorithm, MH-iSAM2 can take multi-mode measurements as inputs and output multi–hypothesis results efficiently, therefore greatly enhance the robustness of SLAM systems.

In the near future, we plan to explore the possibility of MH-iSAM2 as discussed in Sec. VII-D and combine MH-iSAM2 with control or planning modules to improve the overall robustness of real robotic systems.

References

[1] S. Agarwal, K. Mierle, and Others, "Ceres solver," http://ceres-solver.org.

[2] H. Baltzakis and P. Trahanias, "Using multi-hypothesis mapping to close loops in complex cyclic environments," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, May 2006, pp. 824–829.

[3] J. Blanco, J. Fernández-Madrigal, and J. González, "Toward a unified bayesian approach to hybrid metric-topological SLAM," *IEEE Trans. Robotics*, vol. 24, no. 2, pp. 259–270, April 2008.

[4] T. A. Davis, J. R. Gilbert, S. I. Larimore, and E. G. Ng, "A column approximate minimum degree ordering algorithm," *ACM Trans. Math. Softw.*, vol. 30, no. 3, pp. 353–376, Sep. 2004. [Online]. Available: http://doi.acm.org/10.1145/1024074.1024079

[5] F. Dellaert and M. Kaess, "Factor graphs for robot perception," *Foundations and Trends in Robotics*, vol. 6, no. 1-2, pp. 1–139, Aug. 2017.

[6] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM*, vol. 24, no. 6, pp. 381–395, Jun. 1981.

[7] D. Fourie, J. Leonard, and M. Kaess, "A nonparametric belief solution to the Bayes tree," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 2189–2196.

[8] M. Hsiao, E. Westman, G. Zhang, and M. Kaess, "Keyframe-based dense planar SLAM," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, May 2017, pp. 5110–5117.

[9] G. Huang, M. Kaess, J. J. Leonard, and S. I. Roumeliotis, "Analytically-selected multi-hypothesis incremental MAP estimation," in *IEEE Intl. Conf. on Acoustics, Speech and Signal Processing (ICASSP)*, May 2013, pp. 6481–6485.

[10] M. Kaess, A. Ranganathan, and F. Dellaert, "iSAM: Incremental smoothing and mapping," *IEEE Trans. Robotics*, vol. 24, no. 6, pp. 1365–1378, Dec. 2008.

[11] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, "iSAM2: Incremental smoothing and mapping using the Bayes tree," *Intl. J. of Robotics Research*, vol. 31, no. 2, pp. 216–235, 2012.

[12] R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "g2o: A general framework for graph optimization," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, May 2011, pp. 3607–3613.

[13] Y. Latif, C. Cadena, and J. Neira, "Robust loop closing over time for pose graph SLAM," *Intl. J. of Robotics Research*, vol. 32, no. 14, pp. 1611–1626, 2013.

[14] M. Montemerlo and S. Thrun, "Simultaneous localization and mapping with unknown data association using FastSLAM," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, vol. 2, Sept 2003, pp. 1985–1991.

[15] B. Mu, S. Y. Liu, L. Paull, J. Leonard, and J. P. How, "SLAM with objects using a nonparametric pose graph," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, Oct 2016, pp. 4602–4609.

[16] J. Neira and J. D. Tardos, "Data association in stochastic mapping using the joint compatibility test," *IEEE Trans. Robotics and Automation*, vol. 17, no. 6, pp. 890–897, Dec 2001.

[17] T. Pfeifer and P. Protzel, "Robust sensor fusion with self-tuning mixture models," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, Oct 2018.

[18] M. Pfingsthorn and A. Birk, "Generalized graph SLAM: Solving local and global ambiguities through multimodal and hyperedge constraints," *Intl. J. of Robotics Research*, vol. 35, no. 6, pp. 601–630, 2016.

[19] N. Sünderhauf and P. Protzel, "Switchable constraints for robust pose graph SLAM," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, Oct 2012, pp. 1879–1884.

[20] ——, "Towards a robust back-end for pose graph SLAM," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, May 2012, pp. 1254–1261.

[21] S. Tully, G. Kantor, H. Choset, and F. Werner, "A multi-hypothesis topological SLAM approach for loop closing on edge-ordered graphs," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, Oct 2009, pp. 4943–4948.