

*Joint Incremental Inference and Belief Space
Planning for Online Operations of Autonomous
systems*

ELAD I. FARHI

*Joint Incremental Inference and Belief Space
Planning for Online Operations of Autonomous
systems*

RESEARCH THESIS
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

ELAD I. FARHI

SUBMITTED TO THE SENATE OF THE TECHNION - ISRAEL INSTITUTE OF TECHNOLOGY
ADAR, 5781 HAIFA FEBRUAR 2021

This Research Thesis was done under the supervision of Professor Vadim Indelman in the Technion Autonomous Systems Program - TASp

The generous financial help of the Technion is gratefully acknowledged

TO MY SON YOGEV, AND MY BEAUTIFUL WIFE ADA

ACKNOWLEDGMENTS

FIRST AND FOREMOST I AM EXTREMELY GRATEFUL TO MY SUPERVISOR, PROF. VADIM INDELMAN FOR HIS INVALUABLE ADVICE, CONTINUOUS SUPPORT, AND PATIENCE DURING MY PhD STUDY. HIS IMMENSE KNOWLEDGE AND PLENTIFUL EXPERIENCE HAVE ENCOURAGED ME IN ALL THE TIME OF MY ACADEMIC RESEARCH AND DAILY LIFE.

I WOULD LIKE TO THANK DR. ANDREJ KITANOV FOR HIS TECHNICAL SUPPORT AND KIND ADVICE. TO ALL THE MEMBERS IN THE ANPL GROUP AND THE TECHNION AUTONOMOUS SYSTEMS PROGRAM (TASP). IT IS THEIR KIND HELP AND SUPPORT THAT HAVE MADE MY STUDY A TRULY WONDERFUL TIME. TO SIGALIT WHO MADE SURE I HAD EVERYTHING I NEEDED TO SUCCEED, INCLUDING MAKING SURE I TOOK SOME BREAKS FROM TIME TO TIME. TO OMER (A.K.A MY WORK-WIFE), TO URIEL, AND TO MY BRANJA (SHNIKI MOR NOAAM & EREZ), I AM HONORED TO CALL YOU MY FRIENDS, THANK YOU FOR BEING THERE THROUGH THICK AND THIN.

I WOULD LIKE TO EXPRESS MY GRATITUDE AND LOVE TO MY PARENTS. WITHOUT THEIR TREMENDOUS ENCOURAGEMENT, SUPPORT, LOVE AND GUIDANCE, I WOULD NOT BE WHO I AM TODAY. THANK YOU FOR BELIEVING IN ME, EVEN THOUGH I HAVE BEEN SOMEWHAT OF A DIFFICULT CHILD, I HOPE YOU FIND SOME COMFORT IN THE MAN I HAVE BECOME.

FINALLY, TO MY BEST FRIEND, THE MOTHER OF MY CHILD AND THE LOVE OF MY LIFE, ADA. WITHOUT YOUR SUPPORT, LOVE AND UNDERSTANDING IT WOULD BE IMPOSSIBLE FOR ME TO COMPLETE MY STUDY, THIS IS AS MUCH AS YOUR ACHIEVEMENT AS IT IS MINE.

LIST OF PUBLICATIONS

IN PROCEEDINGS

E. I. FARHI AND V. INDELMAN. TOWARDS EFFICIENT INFERENCE UPDATE THROUGH PLANNING VIA JIP-JOINT INFERENCE AND BELIEF SPACE PLANNING. IN IEEE INTL. CONF. ON ROBOTICS AND AUTOMATION (ICRA), MAY 2017.

E. I. FARHI AND V. INDELMAN. IX-BSP: BELIEF SPACE PLANNING THROUGH INCREMENTAL EXPECTATION. IN IEEE INTL. CONF. ON ROBOTICS AND AUTOMATION (ICRA), MAY 2019.

WORKSHOPS

E. I. FARHI AND V. INDELMAN. TEAR DOWN THAT WALL: CALCULATION REUSE ACROSS INFERENCE AND BELIEF SPACE PLANNING. IN TOWARD ONLINE OPTIMAL CONTROL OF DYNAMIC ROBOTS, WORKSHOP IN CONJUNCTION WITH IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION (ICRA), MAY 2019.

UNDER REVIEW

E. I. FARHI AND V. INDELMAN. BAYESIAN INCREMENTAL INFERENCE UPDATE BY RE-USING CALCULATIONS FROM BELIEF SPACE PLANNING: A NEW PARADIGM. ARXIV PREPRINT ARXIV:1908.02002, 2019.

E. I. FARHI AND V. INDELMAN. IX-BSP: INCREMENTAL BELIEF SPACE PLANNING. ARXIV PREPRINT ARXIV:2102.09539, 2021.

PATENTS

E. I. FARHI AND V. INDELMAN & TECHNION RESEARCH AND DEVELOPMENT FOUNDATION LIMITED 2019, EFFICIENT INFERENCE UPDATE USING BELIEF SPACE PLANNING, WO2019171378.

E. I. FARHI AND V. INDELMAN & TECHNION RESEARCH AND DEVELOPMENT FOUNDATION LIMITED 2020, BELIEF SPACE PLANNING THROUGH INCREMENTAL EXPECTATION UPDATING, US20200327358.

Contents

LIST OF PUBLICATIONS	ix
LIST OF FIGURES	xv
LIST OF TABLES	xvii
ABSTRACT	1
ACRONYMS & NOMENCLATURE	4
1 INTRODUCTION	9
1.1 Research Outline	12
1.2 Research Contributions	13
2 LITERATURE SURVEY	15
2.1 The Passive Problem: Inference Given Data	16
2.2 The Active Problem: Decision Making Under Uncertainty and Belief Space Planning	17
2.3 A Unified Perspective for Active and Passive Problems	20
3 INTRODUCING JIP	23
3.1 JIP-Joint Inference and belief space Planning	23
3.2 RUB Inference- Re-Use Belief space planning for inference update	26
3.3 iX-BSP- Incremental eXpectation BSP	27
4 RUBI	31
4.1 Introduction	32
4.2 Inference & BSP Today	34
4.3 Utilizing Inference and BSP similarities	37
4.4 Consistent DA Assumption	43
4.5 Results - Consistent DA	52
4.6 RUBI: Re-Use BSP for Inference update	58
4.7 Results - RUBI	64
4.8 Some Broader Perspective	76
4.9 Concluding RUBI	78

5	INTRODUCING iX-BSP	79
5.1	Related Work	79
5.2	iX-BSP Contributions	81
5.3	BSP Today	84
5.4	Defining the iX-BSP problem	89
5.5	Expectation and the Maximum Likelihood Assumption	90
6	iX-BSP	93
6.1	Comparing Planning Sessions	94
6.2	iX-BSP Overview	96
6.3	Selecting Beliefs for Re-use	99
6.4	Incremental Update of Belief-Tree	104
6.5	Incremental Expectation with Importance Sampling	113
6.6	Results - iX-BSP	120
6.7	The Wildfire Assumption	126
6.8	Results - Wildfire	134
6.9	Some Broader Perspective	138
6.10	Concluding iX-BSP	140
7	iML-BSP	143
7.1	iML-BSP Formulation	143
7.2	Simulation Results - iML-BSP	144
7.3	Real World Experiments - iML-BSP	147
8	CLOSING REMARKS	155
	BIBLIOGRAPHY	158
A	INFERENCE AS A GRAPHICAL MODEL	165
B	DERIVATION OF EQ. (4.14)	167
C	NON-ZEROS IN Q MATRIX	169
D	MULTIPLE IMPORTANCE SAMPLING	175
E	THE $\mathbb{D}_{\sqrt{f}}$ DISTANCE	177
F	PROOF OF THEOREM 1.	179
G	PROOF OF COROLLARY 2.	181
H	PROOF OF THEOREM 2.	183
I	PROOF OF THEOREM 3.	185

J PROOF OF LEMMA 2.	189
K PROOF OF LEMMA 3.	193
L PROOF OF LEMMA 5.	195
M PROOF OF COROLLARY 3.	199
N MEASUREMENT LIKELIHOOD UNDER LINEAR GAUSSIAN MODELS.	201
O GAUSSIAN QUADRATIC AS χ^2	205

List of Figures

3.1.1 Abstract 3D visualization of JIP, addressing both inference and belief space planning under a single process.	24
3.1.2 A fractional 2D visualization of JIP, a novel approach to address both inference and belief space planning under a single process.	25
3.2.1 RUB Inference illustrated on a fractional 2D visualization of JIP.	26
3.3.1 Both RUB Inference and iX-BSP illustrated on a fractional 2D visualization of JIP	28
4.1.1 High level algorithm for RUB Inference presented in a block diagram	32
4.1.2 Illustration for inconsistent DA between planning and succeeding inference	33
4.4.1 Illustration of a Jacobian matrix on its components and dimensions.	49
4.5.1 Inference update method comparison through basic analysis simulation	53
4.5.2 Zoom-in on Inference update method comparison through basic analysis simulation	54
4.5.3 Second simulation under consistent DA assumption, layout and total time	55
4.5.4 Second simulation under consistent DA assumption, per-step timing results	56
4.6.1 The process of incremental DA update, following on iSAM2 methodologies.	62
4.7.1 RUB Inference simulation layout and total time results	65
4.7.2 RUB Inference simulation per-step analysis	67
4.7.3 RUB Inference experiment using the KITTI dataset, layout and total time results .	71
4.7.4 RUB Inference experiment using the KITTI dataset, estimation accuracy analysis .	72
4.7.5 RUB Inference experiment using the KITTI dataset, per-step analysis	73
4.7.6 RUB Inference experiment using the KITTI dataset, computation time analysis .	75
5.2.1 Illustrating the difference between iX-BSP and X-BSP using a simple decision making problem	82
5.5.1 Spatial sensitivity to the ground truth location, the maximum likelihood assumption.	90
6.1.1 Horizon overlap between planning time k and planning time $k+1$	95
6.3.1 X-BSP performs lookahead search on a tree with depth L	99
6.3.2 Illustration of the relative belief distance space	102
6.4.1 The belief update process of iX-BSP presented in a belief tree	106
6.4.2 Illustration for adequate and inadequate representation of a belief by samples	110
6.5.1 Illustration of two consecutive iX-BSP planning sessions	118
6.6.1 High level algorithm for iX-BSP presented in a block diagram	121
6.6.2 Comparing ML-BSP, X-BSP and iX-BSP in a simple simulation	122

6.6.3 Statistical comparison between X-BSP and iX-BSP	125
6.7.1 Illustration of the layout for finding the objective value bounds under wildfire	131
6.8.1 Statistical comparison between iX-BSP with and without wildfire	136
6.8.2 Statistical evaluation of wildfire threshold effect over Objective value	137
7.2.1 Statistical comparison between ML-BSP and iML-BSP	146
7.3.1 Pioneer 3AT robot used for the live experiments	148
7.3.2 The first live experiment stretching across a 35 meter course	150
7.3.3 Planning time results of the first live experiment	151
7.3.4 Number of factors involved in each planning session of the first live experiment	151
7.3.5 The second live experiment stretching across a 148 meter course	152
7.3.6 Planning time results of the second live experiment	153
7.3.7 Number of factors involved in each planning session of the second live experiment .	153
A.1 The relations between Factor grpah, Jacobiam and Bayes Tree	166
C.1 Illustration of matrix non-zeros due to rotations	170
C.2 Illustrating the effectiveness of the bound of non-zeros	173

List of Tables

4.1	Notations for Section 4.4	44
4.1	Notations for Section 4.6.2	60
4.1	Parameters for Section 4.7.2	69
6.1	Notations for Sections 6.2-6.3	98
6.1	Notations for Section 6.4	107
6.1	Notations for Section 6.5	116
6.1	Parameters for Section 6.6.1 following Alg. 3	122
6.2	Parameters for Section 6.6.2 following Alg. 3	124
6.1	Notations for Section 6.7	129
6.1	Parameters for Section 6.8.1 following Alg. 3	135
6.2	Parameters for Section 6.8.2 following X-BSP	137
7.1	Parameters for Section 7.2 following Alg. 3 under iML-BSP	145
7.1	Parameters for Section 7.3 following Alg. 3 under iML-BSP	147

Joint Incremental Inference and Belief Space Planning for Online Operations of Autonomous systems

ABSTRACT

Real life scenarios in Autonomous Systems (AS) and Artificial Intelligence (AI) involve agent(s) that are expected to reliably and efficiently operate online under different sources of uncertainty, often with limited knowledge regarding the environment. These settings necessitate probabilistic reasoning regarding high dimensional problem-specific states. Attaining such levels of autonomy involves two key processes, inference and decision making under uncertainty. The former maintains a belief regarding the high-dimensional state given available information thus far, while the latter, also often referred to as belief space planning (BSP), is entrusted with determining the next best action(s). However, as these problems are computationally expensive, simplifying assumptions or process streamlining are required in order to provide with online or real-time performance. In recent years the similarities between inference and control triggered much work, from developing unified computational frameworks to pondering about the duality between the two. In spite of the aforementioned efforts, inference and control, as well as inference and belief space planning are still treated as two separate processes.

We present in this work the "Joint Inference and Belief Space Planning" (JIP), a novel paradigm that fully utilizes the similarities between probabilistic inference and BSP, thus enabling to re-use computationally expensive calculations. Through the symbiotic relation enabled by JIP we developed new approaches for inference - Ru-Use Belief Inference (RUBI), and for decision making under uncertainty - Incremental eXpectation BSP (iX-BSP). In RUBI we update inference with a precursory planning stage which can be considered as a deviation from conventional Bayesian inference. Rather than updating the belief from the previous time instant with new incoming information (e.g. measurements), RUBI exploits the fact that similar calculations are already performed within planning in order to appropriately update the belief in inference far more efficiently while preserving accuracy.

The iX-BSP approach exploits calculations performed as part of previous planning sessions to efficiently solve a new planning session while accounting for the data that became available since then, which is particularly important while operating in uncertain, potentially dynamically changing, environments.

We demonstrate our novel paradigms on both simulation and real-world data considering active visual SLAM experiments, while benchmarking it against the current top of the line. We show that our paradigms save valuable computation time without introducing simplifying assumptions or affecting accuracy, thus bringing these advanced capabilities more feasible for online setting.

Acronyms & Nomenclature

AI	- Artificial Intelligence
AS	- Autonomous System
SLAM	- Simultaneous Localization and Mapping
DA	- Data Association
MDP	- Markov Decision Process
POMDP	- Partially Observable Markov Decision Process
POMCP	- Partially Observable Monte-Carlo Planning
iSAM	- incremental Smoothing And Mapping
SR / SA	- Single Robot / Agent
MR / MA	- Multi Robot / Agent
RHS	- right hand side
JIP	- Joint Inference and belief space Planning
FG	- Factor Graph
BT	- Bayes Tree
MPC	- Model Predictive Control
PRM	- probabilistic road map
RRT	- rapidly exploring random trees
RRG	- rapidly exploring random graphs
BRM	- Belief road map
RRBT	- rapidly exploring random belief trees
BRL	- Bayesian Reinforcement Learning
DESPOT	- Determinized Sparse Partially Observable Tree
FIRM	- Feedback based Information RoadMap
BSP	- Belief Space Planning
RUBI	- re-use Belief Space Planning for inference update
X-BSP	- expectation Belief Space Planning
ML-BSP	- maximum likelihood Belief Space Planning
iX-BSP	- incremental expectation Belief Space Planning
iML-BSP	- incremental maximum likelihood Belief Space Planning
pdf	- probability density function
ANPL	- Autonomous Navigation and Perception Lab

Notation	Description
$\square_{t k}$	Of time t while current time is k
ΔX_k	State perturbation around linearization point
$\mathcal{M}_{t k}$	Data Association at time t
$A_{t k}$	Jacobian matrix at time t
$b_{t k}$	RHS vector at time t
$\mathcal{A}_{t k}$	Jacobian part related to all factors added at time t
$\mathcal{F}_{t k}$	Jacobian part related to motion factor added at time t
$\mathcal{H}_{t k}$	Jacobian part related to all factors added at time t without the motion factor
$\check{b}_{t k}$	RHS vector related to all factors added at time t
$\check{b}_{t k}^{\mathcal{F}}$	RHS vector related to motion factor at time t
$\check{b}_{t k}^{\mathcal{H}}$	RHS vector related to all factors added at time t without the motion factor
$R_{t k}$	Factorized Jacobian, i.e. square root information matrix
$d_{t k}$	Factorized RHS vector
$A_{t k}^R$	Factorized $\left[R_{t-1 k}^T, \mathcal{A}_{t k}^T \right]^T$
$R_{t k}^{\mathcal{F}}$	Factorized $\left[R_{t-1 k}^T, \mathcal{F}_{t k}^T \right]^T$
$d_{t k}^{\mathcal{F}}$	Factorized $\left[d_{t-1 k}^T, \check{b}_{t k}^{\mathcal{F}T} \right]^T$
$R_{t k}^{aug}$	Factorized Jacobian at time $t-1$ zero padded to match factorized Jacobian at time t
$d_{t k}^{aug}$	Factorized RHS vector at time $t-1$ zero padded to match factorized RHS vector at time t
$Q_{t k}^A$	Rotation matrix for factorizing $A_{t k}$ into $R_{t k}$
$Q_{t k}$	Rotation matrix for factorizing $A_{t k}^R$ into $R_{t k}$
$Q_{t k}^{\mathcal{F}}$	Rotation matrix for factorizing $\left[R_{t-1 k}^T, \mathcal{F}_{t k}^T \right]^T$ into $R_{t k}^{\mathcal{F}}$
$Q_{t k}^{\mathcal{H}}$	Rotation matrix for factorizing $\left[R_{t k}^{\mathcal{F}T}, \mathcal{H}_{t k}^T \right]^T$ into $R_{t k}$
$\mathcal{FG}_{t k}$	Factor graph (FG) at time t
$\mathcal{T}_{t k}$	Bayes Tree (BT) at time t
$\mathcal{M}_{t k}$	Data Association (DA) at time t while current time is k
\mathcal{M}_t^{\cap}	Consistent DA at time t
\mathcal{M}_t^{rmv}	DA at time t from planning inconsistent with inference, indicating factors to be removed
\mathcal{M}_t^{add}	DA at time t from inference inconsistent with planning, indicating factors to be added

Notation	Description
$\{f_r\}_t^{rmv}$	Factors at time t from planning inconsistent with inference, to be removed
$\{f_s\}_t^{add}$	Factors at time t from inference inconsistent with planning, to be added
$\{X\}_t^{inv}$	All states at time t , involved in $\{f_r\}_t^{rmv}$ and $\{f_s\}_t^{add}$
\mathcal{T}_t^{inv}	Sub-BT of $\mathcal{T}_{t k}$ composed of all cliques containing $\{X\}_t^{inv}$
$\{X\}_t^{inv*}$	All states at time t , related to the sub-BT \mathcal{T}_t^{inv}
\mathcal{FG}_t^{inv}	The detached part of $\mathcal{FG}_{t k}$ containing $\{X\}_t^{inv*}$
\mathcal{FG}_t^{upd}	The FG \mathcal{FG}_t^{inv} after DA update
\mathcal{T}_t^{upd}	The sub-BT eliminated from \mathcal{FG}_t^{upd}
$\mathcal{FG}_{t k}^{upd}$	The Factor Graph at time t with all-correct DA
$\mathcal{T}_{t k}^{upd}$	The Bayes Tree at time t with all-correct DA
$\mathcal{M}_{t k}$	Data Association at time t while current time is k
$b[X_{t k}]$	belief at time t while current time is k
$b^-[X_{t k}]$	belief at time $t - 1$ propagated only with action $u_{t-1 k}$
$\mathcal{B}_{k k}$	The entire belief tree from planning at time k
$\tilde{b}[X_{t k}]$	The root of the selected branch for re-use in planning at time t
$\mathcal{B}_{t k}$	The set of all beliefs from planning time k rooted in $\tilde{b}[X_{t k}]$
Dist	The distance between $\tilde{b}[X_{t k}]$ and the corresponding posterior $b[X_{t t}]$
dist	The distance between $b_a^{s'-}[X_{t k}]$ and $b_a^{s-}[X_{t k+l}]$
data	All available calculations from current and precursory planning session
$u_{k:k+L k}^*$	The (sub)optimal action sequence of length L chosen in planning at time k
ε_c	belief distance critical threshold, above it re-use has no computational advantage
ε_{wf}	wildfire threshold, below it distance is considered close-enough for re-use without any update
useWF	a binary flag determining whether or not the wildfire condition is considered
$\mathbb{D}(.)$	belief divergence / metric
$\mathbb{D}^2(.)$	squared $\mathbb{D}(.)$
$\mathbb{D}_{\sqrt{\mathcal{I}}}(p, q)$	The distance between distributions p and q according to the $\mathbb{D}_{\sqrt{\mathcal{I}}}$ distance
$D_{DA}(p, q)$	The divergence between distributions p and q according to the data association difference
Δ	equals $\mathbb{D}^2(b_1^+, b_2^+) - \mathbb{D}^2(b_1, b_2)$, where b_{ip} denotes b_i propagated with motion and measurements

Notation	Description
$b^s[X_{t k+l}]$	The s_{th} sampled belief representing $b[X_{t k+l}]$
$b_a^{s-}[X_{t+1 k+l}]$	The sampled belief $b^s[X_{t k+l}]$ propagated with the a candidate action
$\{b_a^r[X_{t k+l}]\}_{r=1}^n$	A set of n sampled beliefs that are first order children of $b_a^{s-}[X_{t k+l}]$ and are representing $b[X_{t k+l}]$
$b_a^{s'}-[X_{t+i k}]$	A propagated belief from $\mathcal{B}_{t k}$ closest to $b_a^{s-}[X_{t+i k+l}]$
n_u	number of candidate actions per step
$(n_x \cdot n_z)$	number of samples for each candidate action
β_σ	σ acceptance range parameter, for considering measurements as representative
ω_i^n	the weight corresponding to the n_{th} measurement sample for lookahead step i
$q_i(z_{t:i}^g)$	importance sampling distribution at lookahead step i , from which $z_{t:i}^g$ were sampled
$\mathbb{P}(z_{t:i}^\square H, u)$	the nominal distribution at lookahead steps $t : i$
n_i	the number of samples considered at lookahead step i
M_i	number of distributions at look ahead step i from which measurements are being sampled
n_m	the number of measurements sampled from the m_{th} distribution at look ahead step i
$\omega_i(z_{t:i}^{m,g})$	Balance Heuristic likelihood ratio at lookahead step i corresponding to $z_{t:i}^{m,g}$
$\omega_i(z_{t:i}^g)$	private case of $\omega_i(z_{t:i}^{m,g})$ where $m = 1$
$z_{t:i}^{m,g}$	the g_{th} set of future measurements at time instances $t : i$ sampled from the m_{th} distribution
$b^{m,g}[X_{t k+l}]$	the sampled belief representing $b^{m,g}[X_{t k+l}]$ which consider the measurements $z_{k+l+1:t}^{m,g}$
$q_m()$	the m_{th} marginal importance sampling distribution at lookahead step i , $m \in [1, M_i]$
$\{r_i(b[X_{t k}], u)\}_1^j$	j immediate rewards of lookahead step i
$p_i()$	the marginal nominal distribution at lookahead step i
$\tilde{p}_i()$	the nominal distribution at lookahead step i
$\tilde{q}_m()$	the m_{th} importance sampling distribution at lookahead step i , $m \in [1, M_i]$
λ_a	the reward function $a - H\ddot{o}lder$ constant
a	the reward function $a - H\ddot{o}lder$ exponent
$r_{t k}^s$	the immediate reward at lookahead step t , related to $b^s[X_{t k}]$

*Before I refuse to take your questions,
I have an opening statement.*

Ronald Reagan

1

Introduction

FROM ISAAC ASIMOV'S "THREE LAWS OF ROBOTICS"¹, FORMULATED IN OCTOBER 1941, TILL OUR DAYS, AUTONOMOUS SYSTEMS (AS) AND ARTIFICIAL INTELLIGENCE (AI) HAVE COME A LONG WAY. But what is AI? or what makes AS autonomous ? These questions have many answers, stretching across computer sciences as well as philosophy. Although the definition of AS/AI might be multivalent, at their core, regardless of the specific definition you favor, any AS or AI system is required to have two basic processes - inference and decision making. In the following we break-down those two processes to better understand their functionality and importance, using a simple general example of some agent tasked with completing an objective.

The inference problem refers to determining the current state of our agent, given the information it gathered so far and possibly some additional prior knowledge over the agent or the environment. The ability of the agent to infer its current state is crucial in its effort to accomplish the designated objective (whatever it may be). Although the state of the agent is problem sensitive, most real-world problems involve reasoning over high-dimensional state space, e.g. the simultaneous localization and mapping (SLAM) problem dealing with localizing the agent within an unknown environment while

¹First presented in the science fiction short story "Runaround", published by Issac Asimov in March 1942

mapping the same environment. As the state can not always be measured directly, or without any stochastic noise, the reasoning about the state is probabilistic rather than deterministic, i.e. we can only infer a probability over the state of the agent denoted as belief. Solving the aforementioned belief is a synonym to performing inference, or obtaining an estimation of the desired state.

As the inference problem is usually required to be solved sequentially throughout an agents mission, the agent can either consider each instance as a stand-alone problem and solve it from scratch (usually referred to as the batch approach), or to incrementally updated some previous solution with newly acquired information (surprisingly referred to as the incremental approach). As long as there is shared information history between inference sessions, the incremental approach is computationally superior as it only updates some existing solution with new information rather than solving it from scratch. One way to incrementally solve the inference problem is the Bayesian inference approach, in-which Bayes theorem is being used to incrementally update the last posterior belief with newly received information, in order to obtain the new posterior belief.

In addition to the ability to infer its current state, in order to fulfill its objective our agent also requires to decide on its next optimal action. The process of decision making under uncertainty, is tasked with locating the next optimal action given some posterior belief over the current state of the agent. The action optimality is in accordance with its contribution to achieve the desired objective. As discussed in Chapter 2, decision making under uncertainty is an intractable problem, and as such it has and still is the focus of numerous research efforts that try to decrease the computational load by sacrificing performance.

The decision making under uncertainty problem, or more specifically the belief space planning (BSP) approach, locates the optimal action by considering for each candidate action all possible future interactions with the environment (i.e. measurements) given the current posterior belief. In other words, as part of the BSP process, for each possible action, the agent considers all possible future measurements and the corresponding future beliefs.

In the general case, each of the aforementioned future beliefs should be solved in order to provide the prediction of the corresponding future state. In this sense, BSP can be considered as solving numerous inference problems using future measurements.

While the similarities between these two fundamental processes in AI/AS are apparent even before going into their mathematical representations, and in-spite some research effort in this area (see Chapter 2), they are still being treated as two separate processes. As part of our research we came across a built-in inefficiency within these two key processes. The motivation and inspiration for our research is derived from the non-artificial intelligent system we all know - the Human Brain. Although

today, inference and decision-making are treated as two separate processes, the human brain considers them both as part of the same process. In an effort to unlock the symbiosis possibilities between inference and decision making, we present the novel paradigm for Joint Inference and belief space Planning or JIP. In addition to encapsulating both bayesian inference and standard BSP, JIP enables new relations that save valuable computation time without sacrificing accuracy by mimicking similar connections already available in the human brain.

Our research vision is that calculations can be re-used between inference and precursory BSP, and within BSP from different time instances. Our key observation is that inference and BSP have a built-in inefficiency. As part of our research we introduce two new connections to JIP, we will now explain each connection using a simple example. Imagine you have an important business meeting scheduled for tomorrow. You will probably spend the rest of the day, preparing yourself for the meeting, going over all possible questions you might be asked. Tomorrow, during the meeting, if you will encounter a question you prepared yourself for, your response would be instantaneous. In case you will encounter a question you have not prepared yourself for, your response would take longer since you will have to think it through. While this scenario is elementary to every human, this is not how current AS/AI will perform. Autonomous Systems and Artificial Intelligence, as formulated today, will always think of the answer during the meeting, even if it was previously covered during the decision-making (planning) phase. The first novel symbiotic connection we introduce to JIP allows to Re-Use previous Belief space planning session to incrementally update Inference (denoted as RUB Inference), so the agent can incrementally re-use the available answers rather than re-thinking the answers during the meeting.

Now, imagine you have a followup business meeting to prepare for, as before, in order to prepare for the meeting you will go over all possible questions you might be asked. But now, in case similar questions already appear in your old plan, you will re-use them, either as is or by slightly updating them. While this is again elementary for us as humans, an AS/AI would not make use of previous planning sessions for the current plan. The second novel symbiotic connection we introduce to JIP allows to selectively re-use previous planning sessions in order to incrementally create the current plan (denoted as iX-BSP), so the agent can re-use old answers from previous planning sessions.

As part of our research we formulated and tested these new connections both on simulative and real-world data, proving the advantages of the symbiotic relations enabled by JIP over the standard top of the line in inference and BSP. It is worth stressing that these connections do not pose an approximation of the general problem, instead they simply allow to avoid repeating similar calculations across inference and planning.

1.1 RESEARCH OUTLINE

The thesis is organized into 23 chapters in the following manner:

Chapter 2 : surveys the relevant related work on inference and BSP as well as the research efforts on joint inference and planning.

Chapter 3 : introduces the novel concept of joint inference and belief space planning. Starting with the standard inference and BSP in a JIP framework, followed by the new connections for re-using previous planning sessions for inference update (RUB Inference) and for incremental BSP (iX-BSP).

Chapter 4 : introduces the novel concept of RUB Inference. Provides the theoretical background required for the formulation of RUB Inference. We first formulate RUB Inference under a simplifying assumption of consistent data association and test this simplified version in simulation. In the following we relieve this simplifying assumption, and complete the formulation of RUB Inference by formulating how to account for inconsistent data association. We test the full formulation of RUB Inference both in simulation and real-world data. Before concluding this chapter we provide the reader with some broader perspective of RUB Inference.

Chapter 5 : provides an in-depth survey of closely related research efforts. Stating the contributions of iX-BSP the theoretical background required for the formulation of iX-BSP. Defining the iX-BSP problem and providing some insight over the common maximum likelihood assumption in BSP.

Chapter 6 : begins with comparing two planning sessions and discuss the similarities between them as the foundation for iX-BSP. Followed by a complete in-depth overview of iX-BSP, as well as results comparing iX-BSP to the standard X-BSP in simulation. We also introduce a non-integral approximation for iX-BSP denoted as the wildfire assumption, along with its full formulation, as well as both analytical and empirical proofs of the affects wildfire holds over iX-BSP. Before concluding this chapter we provide the reader with some broader perspective of iX-BSP.

Chapter 7 : presents the novel iML-BSP approach. In an effort to demonstrate how iX-BSP can be utilized to benefit existing approximations of the standard BSP paradigm (X-BSP). We present the formulation for iML-BSP as well as testing it both in simulation and live robot experiments.

Chapter 8 : Concluding remarks for our extensive research efforts, along with some possible future research directions.

Chapters A - O : to improve coherence some relevant proofs and derivations are presented in these chapters as appendix.

1.2 RESEARCH CONTRIBUTIONS

In this section we summarize the main contributions of our research.

- We introduce JIP, a novel framework allowing to consider both inference and BSP as part of the same process. We demonstrate how standard inference and BSP are encapsulated within JIP, as well as two new symbiotic connections for re-using BSP for inference update and previous planning sessions .
- We introduce RUB Inference, a novel approach for saving computation time during the inference stage by reusing calculations made during the precursory planning stage
- We provide four exact methods, that utilize our concept under the assumption of consistent DA. We evaluate these four methods and compare them to the state of the art in simulation.
- We provide a paradigm for incrementally updating inconsistent DA, thereby relaxing the aforementioned assumption.
- We evaluate the RUB Inference paradigm and compare it to the state of the art both in simulation and on real-world data, considering the problem of autonomous navigation in unknown environments.
- We present a novel paradigm for incremental expectation belief space planning with selective resampling (iX-BSP). Our approach incrementally calculates the expectation over future observations by a set of samples comprising of newly sampled measurements and re-used samples generated at different planning sessions.
- We identify the problem of iX-BSP with selective resampling as a Multiple Importance sampling problem, and provide the proper formulation while considering the balance heuristic.
- We evaluate iX-BSP in simulation and provide statistical comparison to X-BSP, which calculates expectation from scratch, while considering the problem of autonomous navigation in unknown environments, across different randomized scenarios.
- We introduce the wildfire approximation into iX-BSP, which enables one to controllably trade accuracy for performance.
- We provide an analytical proof of the affect the choice of a wildfire threshold would have over the objective value, in the form of bounds over the objective value error.

- We provide empirical results of using wildfire within iX-BSP, as well as the affect wildfire holds over the objective value error.
- We support our claim, that iX-BSP can be used to improve approximations of the general problem of X-BSP, by introducing to iX-BSP the commonly used ML approximation, and denote it as iML-BSP. The novel approach of iML-BSP, incrementally calculates the expectation over future observations, while considering either the most likely observation or some previously sampled observation, given from a precursory planning session.
- We evaluate iML-BSP in simulation as well as in real-world experiments and compare it to the commonly used approximation for the X-BSP problem, ML-BSP, while considering the problem of autonomous navigation in unknown environments and active visual-SLAM setting with belief over high dimensional state space.

*Get your facts first,
then you can distort them as you please.*

Mark Twain

2

Literature Survey

AUTONOMOUS SYSTEMS AND ARTIFICIAL INTELLIGENCE RELATED PROBLEMS, CAN BE DIVIDED INTO TWO CATEGORIES, the passive problem, usually referred to as estimation, inference or perception and the active problem, usually referred to as control, planning, decision making under uncertainty, or active perception.

Since a deterministic scheme can only provide limited representation for the real life passive and active problems (e.g. a deterministic model for real outdoor environments is not feasible), both passive and active problems often involve maintaining and reasoning about a probability density function (pdf), also known as the *belief*, over the state of interest conditioned on available information. Although belief representation is more expensive computationally when compared to deterministic models, it serves a better fit to real world scenarios. The current research effort in the field of AS and AI is twofold: coming up with paradigms that would better fit real life scenarios and streamlining those methods to achieve real-time performance. Our research proposal deals with both aforementioned aspects, while inspired by *Mother Nature*.

2.1 THE PASSIVE PROBLEM: INFERENCE GIVEN DATA

The inference problem, entrusted with maintaining a belief over variables of interest (e.g., robot poses) given available information and robot actions, has been addressed by the research community extensively over the past decades. In particular, focus was given to inference over high-dimensional state spaces, with Simultaneous Localization And Mapping (SLAM) being a representative problem, and to computational efficiency to facilitate online operation, as required in numerous robotics systems. Over the years, the solution paradigm for the inference problem has evolved. From EKF based methods (e.g. [12], [27]), through information form recursive (e.g. [68]) and smoothing methods (e.g. [14], [16])), and in recent years up to incremental smoothing approaches, such as iSAM [33] and iSAM2 [36].

The inference problem is naturally represented using graphical models such as factor graph [44] and Bayes Tree (BT) [35]. This representation enabled more efficient paradigms as will be reviewed later on. Figure A.1 presents the connections between those different representations that are exploited by incremental smoothing approaches such as iSAM2 [36]. The factor graph ((a) in Figure A.1) encodes a joint probability density function (pdf) for a full-SLAM problem (state is composed out of poses and landmarks), or equivalently for Gaussian distributions, the original non-linear least squares problem. On the other hand, BT represents a factorization of the joint pdf in terms of conditionals for a given variable elimination order. For Gaussian distributions, BT efficiently represents the square root information matrix R and the RHS vector d , where each clique over a subset of variables encodes non-zero entries in appropriate sub-blocks and entries of R and d , see illustration in (c) and (d) in Figure A.1. When new information is received, the iSAM2 algorithm efficiently updates only the impacted parts in BT, an operation that corresponds to updating an existing factorization rather than calculating a new one from scratch.

The incremental approach of iSAM and iSAM2, suggests updating the existing belief estimation with new acquired measurements. While iSAM can only update linear systems incrementally and requires periodic batch steps for reordering and re-linearization, iSAM2 offers a fully incremental algorithm for nonlinear least-squares (NLS) problems when both variable reordering and re-linearization are being done incrementally.

2.2 THE ACTIVE PROBLEM: DECISION MAKING UNDER UNCERTAINTY AND BELIEF SPACE PLANNING

Given the posterior belief from the inference stage, decision making under uncertainty and Belief Space Planning (BSP) approaches are entrusted with providing the next optimal action sequence given a certain objective. The aforementioned is accomplished by reasoning about belief evolution for different candidate actions while taking into account different sources of uncertainty. The corresponding problem is an instantiation of a partially observable Markov decision process (POMDP) problem, known as PSAPCE-complete [55], hence computationally intractable for all but the smallest problems, i.e. no more than a few dozen states [32].

The main cause for the BSP problem intractability, resides with the use of expectation in the objective function, i.e. reasoning about belief evolution along different candidate actions while considering all possible future measurements

$$J(\mathcal{U}) = \mathbb{E}_z \left\{ \sum_i r_i(b_i, u_{i-1}) \right\}. \quad (2.1)$$

The objective over a candidate action sequence \mathcal{U} , is obtained by calculating the expected value of all possible rewards (costs) r received from following \mathcal{U} . Since the reward (cost) function is a function of the belief b and the action led to it u , in practice the objective considers all future beliefs obtained from following \mathcal{U} , i.e. the expectation considers the joint measurement likelihood of all future measurements z . We refer to this general problem as the full solution of BSP, denoted by X-BSP, eXpectation based BSP.

The exponential growth of possible measurements and candidate actions, usually denoted as the *curse of history*, is the key aspect targeted by a lot of research efforts. Performing inference over multiple future beliefs is the key reason for the costly computation time of X-BSP. In a planning session with a horizon of 3 steps ahead, 3 candidate actions per step and 3 samples per action, we are required to solve a staggering number of 819 beliefs. Cutting down on the computation time of each belief would benefit the overall computation time of the planning process.

As in any computational problem, one can either streamline the solution process or change the problem, i.e. take simplifying assumptions or approximations.

Indeed, over the years, numerous approaches have been developed that trade-off suboptimal performance with reduced computational complexity of POMDP, see e.g. [28, 45, 57, 70]. While the majority of these approaches, including [5, 58, 59, 73], assumed some sources of absolute information

(GPS, known landmarks) are available or considered the environment to be known, recent research relaxed these assumptions, accounting for the uncertainties in the mapped environment thus far as part of the decision making process [29, 40] at the price of increased state dimensionality.

Other than assuming available sources of absolute information, some approaches use discretization in order to reduce computational complexity. Sampling based approaches, e.g. [1, 5, 28, 59], discretize the state space using randomized exploration strategies to locate the belief's optimal strategy. Other approaches, e.g. [61], discretize the action space thus trading optimality with reduced computational load. While many sampling based approaches, including probabilistic roadmap (PRM) [39], rapidly exploring random trees (RRT) [47], RRT* [38] and rapidly exploring random graph (RRG) [37] assume perfect knowledge of the state (i.e. MDP framework) along with deterministic control and known environment, efforts have been made to assuage these simplifying assumptions. These efforts vary in the alleviated-assumptions, from the belief roadmap (BRM) [59] and the rapidly exploring random belief trees (RRBT) [5], through, Partially Observable Monte-Carlo Planning (POMCP) [64], Determinized Sparse Partially Observable Tree (DESPOT) [49, 76] and up to active full SLAM in discrete [65] and continuous [29] domains accounting for uncertainties in the environment mapped thus far as part of the decision making process (e.g. [29, 40, 65, 72]) at the price of increased state dimensionality.

While all the aforementioned research efforts tackle the curse of history through providing various approximations to the X-BSP problem, a common denominator for some of them is the Maximum Likelihood (ML) assumption [58], which allows to prune X-BSP by considering only the maximum likelihood measurements rather than all possible ones. We denote the use of ML in BSP as ML-BSP.

Other aspects of BSP have also been addressed in recent years, one of which is informative planning. Informative planning is the problem of choosing an action sequence that maximizes the sensor information obtained about an underlying field of interest, e.g. [4]. The aforementioned field of interest could be anything from minimizing the estimation uncertainty [28], through refining the environment model [51] and up to streamlining perception measurements' quality [8].

Another aspect that has been addressed in the context of decision making under uncertainty is unknown environments. Since creating or obtaining an accurate probability distribution of an unknown environment is unfeasible, research efforts has been made in order to overcome this difficulty, while the majority of them turn to Reinforcement Learning (RL) or for the POMDP case, Bayesian Reinforcement Learning (BRL) [22].

For example, [61] offers high speed navigation POMDP planner based on Machine Learning in order to categorize obstacles beyond the known regions of the map. Using the movement history as a training set along with an offline calculated database of discretized possible actions for different velocities, [61] predicts collision probabilities in uncharted territories of the environment online.

Recent research has been trying to streamline the planning paradigm itself, in parallel to simplifying assumptions in order to save computation time. An example for such work is given by [2, 3], which are extensions of [1], offering feedback-based information roadmap or FIRM, a multi-query approach for planning under uncertainty, which is a belief space variant of probabilistic roadmap methods. FIRM offers to streamline inefficiencies in the form of similar repeating calculations, found in Model Predictive Control (MPC) paradigm, by performing some of the calculations offline and utilizing them in the expense of accuracy and optimal performance. Under the assumption of a known environment (i.e. not high dimensional state POMDP), FIRM samples the belief space offline using PRM, where each node is a belief. The covariance of each belief is defined by considering only few of the possible paths to this node, thus trading-off the dependency over history with accuracy. These sampled beliefs are represented in a graph while the costs associated with each are independent from one another. The purpose of FIRM is to use those offline pre-calculated solutions of sampled beliefs during planning. It does so by forcing the current belief to reach a sampled belief at the end of each horizon. In addition to sacrificing accuracy, this paradigm dismisses Data Association (DA) aspects as measurement noise and is not applicable for a Multi Robot (MR).

Another example for such work is [7], which offers to reduce computational complexity by eliminating redundant computations between similar candidate action sequences *from the same horizon*. Using a BT representation along with iSAM2 efficient update methodologies, belief evolution along candidate actions is being done incrementally thus saving valuable computation time.

A crucial component in both inference and BSP is data association (DA), i.e. associating between sensor observations and the corresponding landmarks. Incorrect DA in inference or BSP can lead to catastrophic failures, due to wrong estimation in inference or incorrect belief propagation within BSP that would result in incorrect, and potentially unsafe, actions. Recent research thus focused on developing approaches that are robust to incorrect DA, considering both passive [6, 30, 54, 66] and active perception [56].

2.3 A UNIFIED PERSPECTIVE FOR ACTIVE AND PASSIVE PROBLEMS

Regardless of the decision making approach being used, in order to determine the next (sub)optimal actions the current belief is propagated using various action sequences. The propagated beliefs are then solved in order to provide an objective function value, thus enabling to determine the (sub) optimal actions. Solving a propagated belief is equivalent to performing inference over the belief, hence solving multiple inference problems is inevitable when trying to determine the (sub) optimal actions.

Obvious similarities between inference and decision making, triggered much work in recent years. For example, Kobilarov et al. [41] and Ta et al. [67] developed Differential Dynamic Programming (DDP) and Factor Graph (FG) based unified computational frameworks, respectively, for inference and decision making. Toussaint and Storkey [71] provided an approximate solution to the Markov Decision Process (MDP) problem using inference optimization methods, and Todorov [69] investigated the duality between optimal control and inference for the MDP case. Both Toussaint and Storkey [71] and Todorov [69] refer to the MDP case, rather than the more general case where the state is partially observable and thus has to be inferred probabilistically within a POMDP framework. Interestingly, be the decision making approach as it may, it has to solve numerous inference problems in order to determine the (sub)optimal actions. However, despite the aforementioned research efforts, inference and BSP are still handled as two separate processes.

Our work revolves around finding a unified paradigm for both inference and decision making under uncertainty within a POMDP framework while considering unknown environment. We believe this unified framework would help streamline both inference and BSP as we would show in the following chapters.

Thriving for a unified paradigm for both inference and decision making can get validation when considering the ideal AS/ AI provided by Mother Nature - the *Human Brain*. Making use of biological models, even from brain sciences, is not an uncharted territory in AS and AI, e.g. [53] made use of rodent hippocampus model in order to create RatSLAM, a passive vision-only SLAM paradigm.

Interestingly enough, inference and decision making under uncertainty in the human brain are tightly entwined, hence it provides motivation to pursue an equivalent paradigm for AS and AI. For example, as discussed in [60, 62, 63], simulating possible future events in the human brain is done by the same mechanism used to remember the past. Both regions of the Medial Temporal Lobe (MTL), i.e. the hippocampus and the parahippocampal cortex, are engaged when individuals build simulations of events located in the future, past or present. Moreover, amnesia patients have shown symmetry between their ability to recall past events and compose future events, i.e. severely amnesic

patients who couldn't remember any specific episodes from their past couldn't envision any specific episodes in their future, although having no physical damage to their MTL. These findings validate our research ideas and provide us with a conceptual foundation that a joint inference and planning approach would prove to be more efficient.

*The distinction between past, present and future
is only a stubbornly persistent illusion.*

Albert Einstein

3

Introducing JIP

IN THIS CHAPTER WE PRESENT THE JOINT INFERENCE AND BELIEF SPACE PLANNING APPROACH, OR JIP. We start by describing the standard inference and BSP under JIP (Section 3.1), and continue with presenting two paradigms that unlock the potential of JIP: RUB Inference (Section 3.2) allowing for efficient inference update through the re-use of BSP calculations , and iX-BSP (Section 3.3) allowing to perform BSP through incrementally updating previous planning sessions. Here we shortly introduce the main concept of RUB Inference and iX-BSP, but in the following chapters we dive-into each of which and provide a thorough formalization analysis and validation.

3.1 JIP-JOINT INFERENCE AND BELIEF SPACE PLANNING

The Joint Inference and belief space Planning (JIP) approach, considers both inference and decision making under uncertainty as part of a single process. In this section we describe how both inference and BSP can be considered as part of the same process.

Figure 3.1.1 provides an abstract 3D-graph visualization of JIP. Each node in the graph denotes either a posterior (blue sphere) or predicted (silver sphere) belief, and each edge denotes the action and

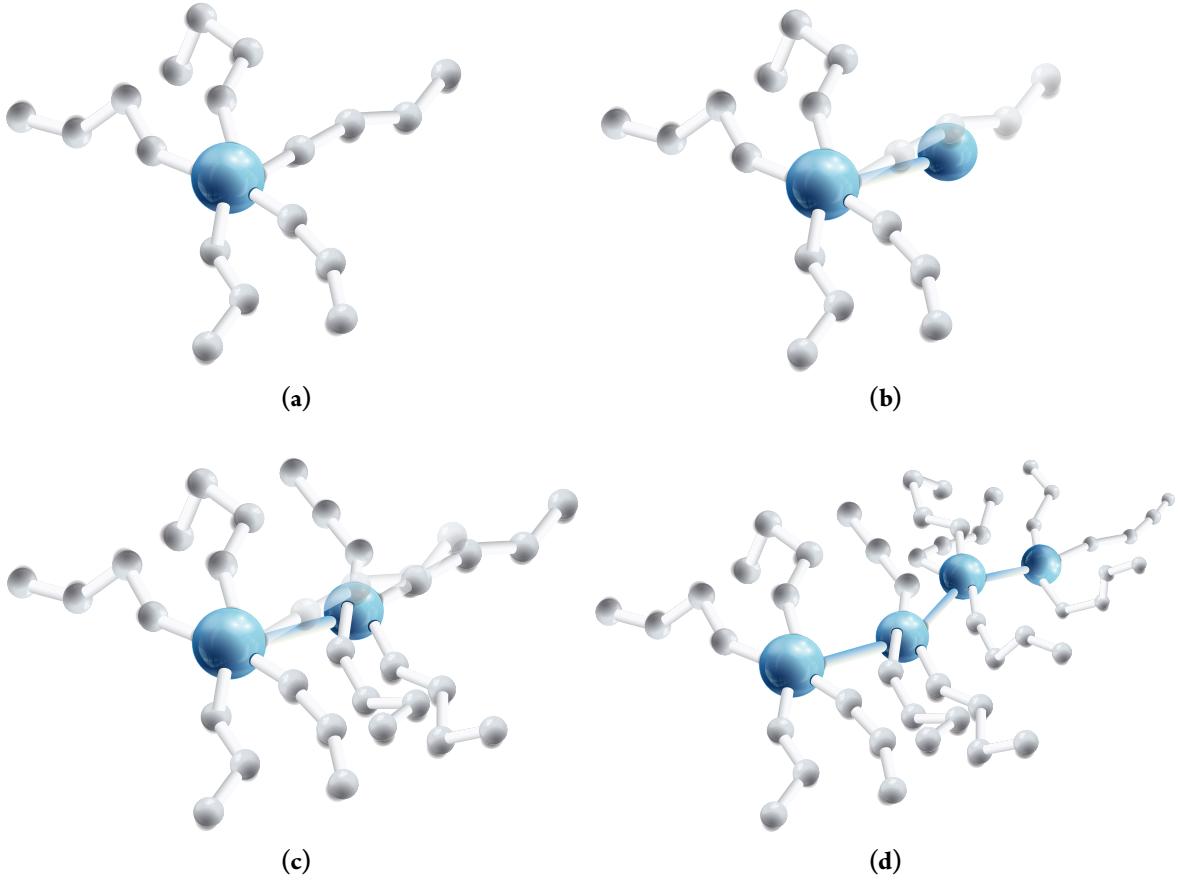


Figure 3.1.1: Abstract 3D visualization of JIP, addressing both inference and belief space planning under a single process. Nodes represent posterior and future beliefs denoted by blue and silver respectively. Edges represent actions and corresponding measurements, either candidate actions with sampled measurements in white or executed actions and actual measurements in light blue. To avoid clutter only few candidate action sequences are presented for each belief, and for each action sequence only a single belief instantiation. In (a), a single posterior belief (blue) with 5 candidate action sequences branching out of it, each with appropriate future beliefs (silver). After executing the action selected in (a) (denoted as watermark in (b)), the obtained measurements are used to calculate the new posterior belief as illustrated in (b). (c) considers 4 candidate action sequences for the new posterior belief. (d) Illustrating a plan-act-infer system with two more executed steps.

measurement that led to it. We will now explain how Figure 3.1.1d represents a plan-act-infer framework using Figures 3.1.1a-3.1.1c. Consider the single blue node in Figure 3.1.1a, as representing probability distribution over the current joint state (i.e. belief). Under a plan-act-infer framework we will need to plan for the next optimal action sequence. We examine several candidate action sequences, by sampling measurements and propagating beliefs along each of which. Those future beliefs along candidate action sequences are denoted respectively by silver nodes and white edges branching out of our blue node. The length of each tentacle, i.e. the number of silver nodes, denote the planning horizon of the appropriate candidate action sequence. For each candidate action sequence we can calculate a reward (cost) value, thus eventually choosing the best candidate action as the one with

the maximum reward value. We then execute this action and receive measurements (both denoted by the light-blue edge in Figure 3.1.1b), and calculate the next posterior belief, i.e. the newly added blue node. The chosen candidate action sequence is denoted by a watermark in Figure 3.1.1b. Given the newly calculated posterior belief, Figure 3.1.1c considers candidate action sequences and propagate future beliefs. We are now in position to understand Figure 3.1.1d, illustrating inference and BSP over a single graph. In order to avoid clutter only few candidate action sequences are considered for each planning session (i.e. the degree of each node), for each candidate action only a single predicted belief is considered, and all selected action sequences (e.g. watermarks in Figures 3.1.1b-3.1.1c) were pruned.

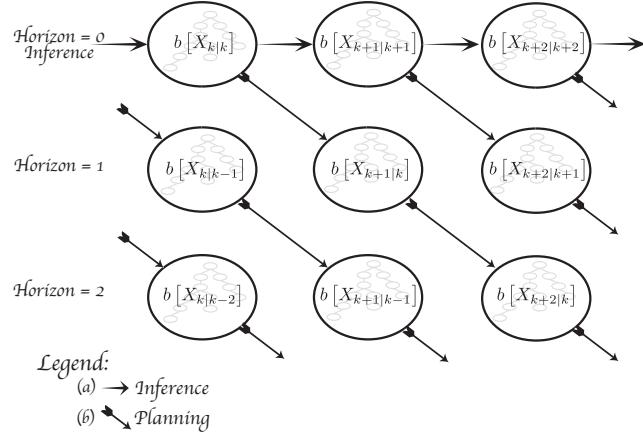


Figure 3.1.2: A fractional 2D visualization of JIP, a novel approach to address both inference and belief space planning under a single process. Here, $b[X_{k+1|k}]$ stands for the belief of the joint state in time instance $k + 1$ while current time is k and each row stands for a different planning horizon. The relations between different beliefs in the graph are denoted by different arrows. (a) Inference; (b) Planning step.

As Figure 3.1.1d might be too abstract to convey how inference and BSP are encapsulated within JIP, we provide its atrophied 2D visualization in Figure 3.1.2. Each node in Figure 3.1.2 represents a belief, i.e. $b[X_{k+1|k}]$ denotes the joint belief of state X at a future time instant $k + 1$ given that the current time is k . Adding the subscript $\square_{t|k}$ denoting current time k , allows us to consider past present and future related parameters using the same unified notation. The right facing arrows i.e. (a) in Figure 3.1.2 denote inference at sequential time instances, corresponding to the light-blue edges in Figure 3.1.1d. From each node there is a possible planning action distinguished by different possible controls. The optimal planning action is denoted as a diagonal arrow in the 2D representation i.e. (b) in Figure 3.1.2, corresponding to the watermarks in Figures 3.1.1b-3.1.1c. Each row in Figure 3.1.2 represents a different planning horizon step. While the first row denotes inference (corresponding to the blue nodes in Figure 3.1.1d), each of the others denote future beliefs (corresponding to the silver nodes in Figure 3.1.1d). Each column in Figure 3.1.2 contains beliefs which reason about the

same time instance, but the deeper we go down the column, the older the information this reasoning is based upon. For example, $b[X_{k+1|k+1}]$, $b[X_{k+1|k}]$ and $b[X_{k+1|k-1}]$ all denote the joint belief of state X at time instant $k+1$ but for different current time instances.

Although Figure 3.1.2 visualize only a fraction of JIP as presented in Figure 3.1.1d, it is adequate for demonstrating some of the re-use potential enabled through JIP, as will be discussed in the following sections.

3.2 RUB Inference- Re-USE BELIEF SPACE PLANNING FOR INFERENCE UPDATE

In this section we present a paradigm which utilizes JIP in an effort to improve inference update. Introducing one of our main contributions, Re-Use Belief space planning for Inference update, denoted as RUB Inference. Instead of updating the last posterior belief with current information,

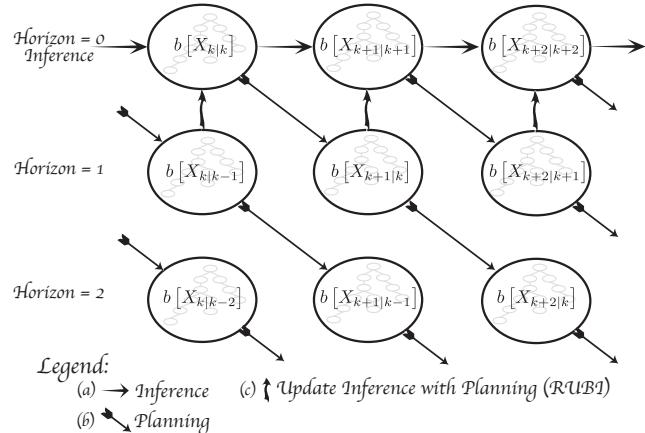


Figure 3.2.1: A fractional 2D visualization of JIP, a novel approach to address both inference and belief space planning under a single process. Here, $b[X_{k+1|k}]$ stands for the belief of the joint state in time instance $k+1$ while current time is k and each row stands for a different planning horizon. The relations between different beliefs in the graph are denoted by different arrows. (a) Inference; (b) Planning step; (c) Updating Inference with precursory planning (Chapter 4).

under RUB Inference we update some previously calculated belief, given from a previous planning session. The advantage of not necessarily using the last posterior for inference update is that there might be some belief, available from previous planning session, that would require less information to update. To better understand how RUB Inference fits within a plan-act-infer framework we provide a high-level walkthrough using Figure 3.2.1. Let us assume we are at current time k , we perform BSP given the currently available posterior belief $b[X_{k|k}]$ in an effort to find the optimal planning control, denoted by the diagonal arrows in Figure 3.2.1 or watermarks in Figures 3.1.1b-3.1.1c.

After executing the optimal action and receiving new measurements, under standard bayesian inference update the posterior $b[X_{k|k}]$ should be updated with the aforementioned in order to obtain

the new posterior $b[X_{k+1|k+1}]$. Inference update under the standard Bayesian inference is denoted by the right facing arrows in Figure 3.2.1, equivalent to the light-blue edges in Figure 3.1.1. One of our key observations is that there might be an available belief closer to the desired $b[X_{k+1|k+1}]$ than $b[X_{k|k}]$, and so updating it instead of $b[X_{k|k}]$ will result with the same posterior belief $b[X_{k+1|k+1}]$ but with a potentially reduced computational load. Although RUB Inference might be considered as a deviation from the standard Bayesian inference, it is not an approximation. The reduced computational effort is originated from needing to update less information in order to obtain the desired posterior belief.

The upward facing arrows in Figure 3.2.1, illustrate RUB Inference when considering one of the previously calculated beliefs $b[X_{k+1|k}]$ from planning at time k , that considers the optimal action, as the closest to the desired $b[X_{k+1|k+1}]$.

By closest we mean

$$b[X_{k+1|k}] = \arg \min_{b \in \mathcal{B}} \mathbb{D}(b, b[X_{k+1|k+1}]) \quad (3.1)$$

where \mathcal{B} denotes the set of all previously calculated available beliefs, and \mathbb{D} is some belief distance. It is worth stressing that the set \mathcal{B} of candidate beliefs for re-use need not be necessarily from the precursory planning session or to contain only beliefs with the same action sequence as $b[X_{k+1|k+1}]$.

In Chapter 4 we formalize RUB Inference as well as putting it to the test using both simulative and real-world data assuming a visual SLAM navigation problem in an unknown environment and high-dimensional state space.

3.3 iX-BSP- INCREMENTAL eXPECTATION BSP

In this section we present a paradigm which utilizes JIP in an effort to improve the planning process. Introducing another one of our main contributions, Incremental eXpectation BSP, denoted as iX-BSP. Considering the full un-approximated problem we denoted as X-BSP, we selectively re-use beliefs, previously calculated in precursory planning sessions, in order to incrementally create the current planning tree. As belief propagation is computationally expensive, we can save valuable computation time by incrementally updating previously calculated beliefs with relevant information. It is worth stressing that iX-BSP is not an approximation to X-BSP, and bares no integral assumptions other than access to previous planning sessions.

To better understand how iX-BSP fits within a plan-act-infer framework we provide a high-level walkthrough using Figure 3.3.1. Let us assume we are at current time k , we perform BSP given the currently available posterior belief $b[X_{k|k}]$ in an effort to find the optimal planning control, denoted

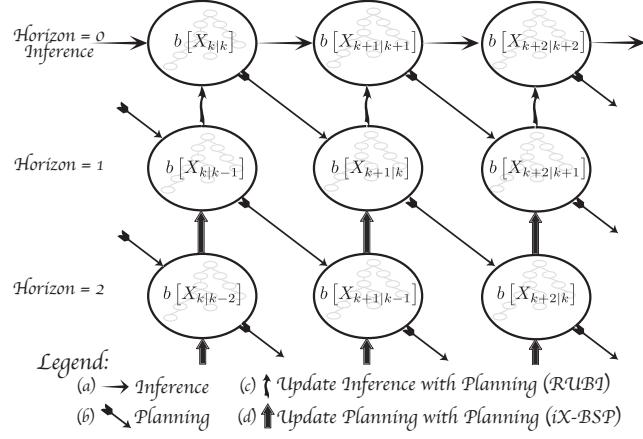


Figure 3.3.1: A fractional 2D visualization of JIP, a novel approach to address both inference and belief space planning under a single process. Here $b[X_{k+1|k}]$ stands for the belief of the joint state in time instance $k+1$ while current time is k and each row stands for a different planning horizon. The relations between different beliefs in the graph are denoted by different arrows. (a) Inference; (b) Planning step; (c) Updating Inference with precursory planning (Chapter 4); (d) Update planning with precursory planning (Chapters 5-6).

by the diagonal arrows in Figure 3.3.1 or watermarks in Figures 3.1.1b-3.1.1c. Under X-BSP, we calculate the reward of each candidate action sequence by considering the cumulative rewards along the action sequence. For the general case of belief based reward, this will require propagating and solving beliefs along the planning horizon. This propagation is represented by the diagonal arrows in Figure 3.3.1 or the white edges connected to the blue nodes in Figure 3.1.1.

The upward facing arrows denoted by (d) in Figure 3.3.1, illustrates iX-BSP under JIP. By adding the upward facing arrows denoted by (d) in Figure 3.3.1, we can now generalize The upward facing arrows in Figure 3.3.1 denote updating some future belief, i.e. belief from horizon larger than zero, with new information, e.g. new measurement values and data association (DA). When updating future belief we actually update the current time this belief is based upon, e.g. when updating $b[X_{k+1|k-1}]$ with information from time step k , denoted in Figure 3.3.1 by (d), we obtain $b[X_{k+1|k}]$. Updating belief from horizon of one is equivalent to the inference update stage, e.g. through (c) in Figure 3.3.1 $b[X_{k+1|k}]$ is updated to $b[X_{k+1|k+1}]$.

And so, instead of propagating future beliefs along the diagonal line of Figure 3.3.1 for each candidate action sequence, we incrementally update beliefs from previous planning sessions with current information. In-spite of the way it is illustrated in Figure 3.3.1, the beliefs chosen for update do not necessarily need to be from precursory optimal action sequences or referring to the same future time step. Potentially, any previously calculated belief can be incrementally updated, assuming it is close enough to current information so the update is worth while.

In Chapters 5-6 we formalize iX-BSP as well as putting it to the test assuming a visual SLAM navigation problem in an unknown environment and high-dimensional state space. As iX-BSP is equiv-

alent to the un-approximated X-BSP problem, in Chapter 7 we demonstrate how it can be utilized also over approximations of X-BSP. Considering the common maximum likelihood (ML) approximation of X-BSP denoted as ML-BSP, we utilize the iX-BSP paradigm and provide iML-BSP.

*Common sense is actually nothing more than
a deposit of prejudices laid down in the mind
prior to the age of eighteen.*

Albert Einstein

4

RUBI

THROUGH THE SYMBIOTIC RELATION ENABLED BY CONSIDERING THE JOINT INFERENCE AND BSP PROBLEMS WE MAKE THE FOLLOWING KEY RESEARCH HYPOTHESIS: INFERENCE CAN BE EFFICIENTLY UPDATED USING A PRECURSORY PLANNING STAGE. In this chapter we investigate this novel concept for inference update using BSP, considering operation in uncertain or unknown environments and compare it against the current state of the art in both simulated and real-life environments. This chapter is organized as follows. Section 4.1 introduces the concept of RUB Inference and reviews the contributions of this chapter. Section 4.2 reviews current formulation of a plan-act-infer system, focusing on inference update. Section 4.4 presents RUB Inference under the simplifying assumption of consistent data association. Section 4.5 presents a thorough analysis of the aforementioned and a comparison to related work. Section 4.6 relieves the simplifying assumption of consistent DA and presents RUB Inference while accounting for inconsistent DA. Section 4.7 presents a thorough analysis of RUB Inference and a comparison to related work both on simulation and real-world data. Section 4.8 discusses a broader perspective of RUB Inference. Section 4.9 captivates the conclusions of this chapter along with possible extensions and usage. To improve coherence, several aspects are covered in appendices.

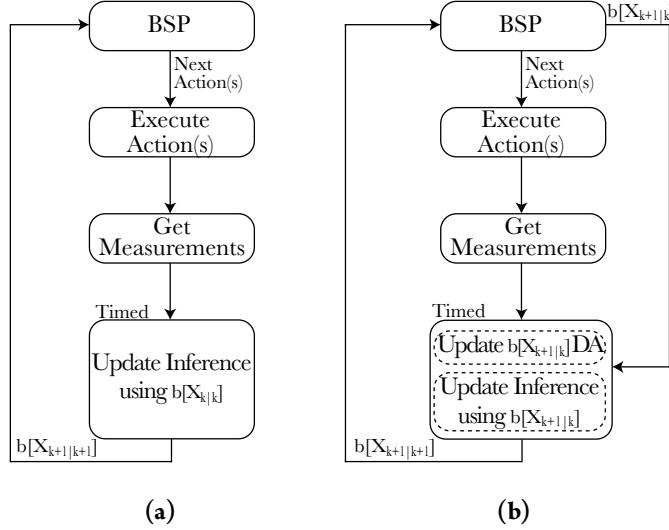


Figure 4.1.1: High level algorithm for joint inference and BSP presented in a block diagram: (a) presents a standard plan-act-infer framework with Bayesian inference and BSP treated as separate processes; (b) presents our novel approach for inference update using precursory planning. Instead of updating the belief from precursory inference with new information we propose to update the belief from a precursory planning phase. Since the only difference between (a) and (b) manifests in computation time within the inference block, it is timed for comparison.

4.1 INTRODUCTION

Updating inference with a precursory planning stage can be considered as a deviation from conventional Bayesian inference. Rather than updating the belief from the previous time instant with new incoming information (e.g. measurements), we propose to exploit the fact that similar calculations have already been performed within planning, in order to appropriately update the belief in inference more efficiently. We denote this novel approach by Re-Use BSP for inference, or RUB Inference in short.

The standard plan-act-infer framework of a typical autonomous system with conventional Bayesian approach for inference update is presented in Figure 4.1.1a. First, BSP determines the next best action(s) given the posterior belief at current time; the robot performs this action(s); information is gathered and the former belief from the precursory inference is updated with new information (sensor measurements); the new posterior belief is then transferred back to the planning block in order to propagate it into future beliefs and provide again with the next action(s).

Our proposed concept, RUB Inference, is presented in Figure 4.1.1b. RUB Inference differs from the conventional Bayesian inference in two aspects: The output of the BSP process and the procedure of inference update. As opposed to standard Bayesian inference, in RUB Inference, BSP output includes the next action(s) as well as the corresponding propagated future beliefs, no other

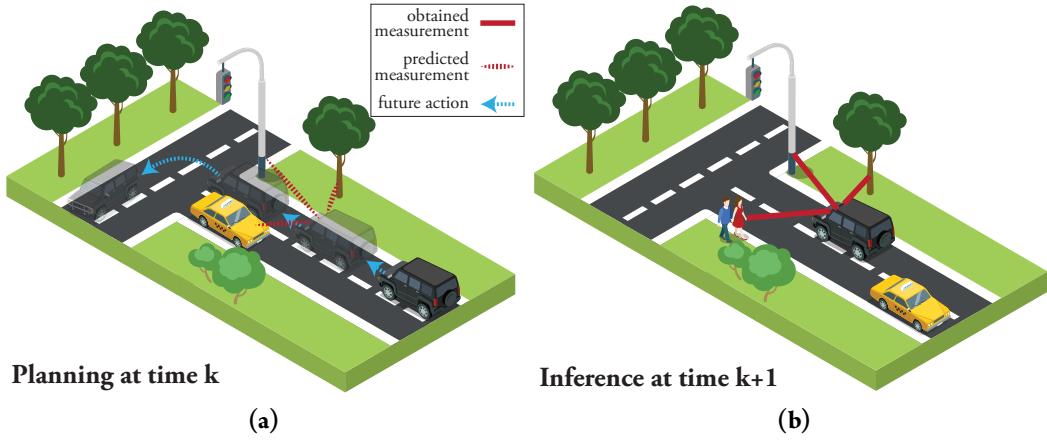


Figure 4.1.2: Illustration for inconsistent DA between planning and succeeding inference: (a) at time k , our robot (i.e. the black jeep) plans three steps into the future. For the future step $k + 1$ it predicts measurements from three landmarks (tree, traffic-light and taxi). (b) After executing the first action our robot obtained three measurements from the environment. Two of them (i.e. tree and traffic-light) match the predicted DA from precursory planning session, while the third is associated to a new landmark (i.e. the couple that came out of the taxi).

changes are required in BSP in order to facilitate RUB Inference. These beliefs are used to update inference while potentially taking care of data association aspects, rather than using the belief from precursory inference as conventionally done under Bayesian inference. As can be seen in Figure 4.1.1b, the inference block contains data association (DA) update before the actual inference update. There are a lot of elements that can cause the DA in planning to be partially different than the DA established in the successive inference, e.g. estimation errors, disturbances, and dynamic or un-modeled unseen environments.

We start investigating this novel concept under a simplifying assumption that the DA considered in planning is consistent to that acquired during the succeeding inference, e.g. we predicted an association to a specific previously mapped landmark and later indeed observed that landmark. Since data association only relates to connections between variables and not to the measurement value, we are left with replacing the (potentially) incorrect measurement values, used within planning, with the actual values. Under this assumption, we provide four exact methods to efficiently update inference using the belief calculated by the precursory planning phase. As will be seen, these methods provide the same estimation accuracy as the conventional Bayesian inference approach, with a significantly shorter computation time.

We later relax the simplifying assumption mentioned above, and show inference can be efficiently updated using the precursory planning stage even when the DA considered in the two processes is partially different. Figure 4.1.2 illustrates such a case of inconsistent DA using a simple navigation problem. At time k our automated car (denoted by a black jeep), performs planning with a horizon

of three steps. Figure 4.1.2a presents the chosen candidate action sequence along with the predicted measurements for future time $k+1$. Our automated car predicts that at future time $k+1$ it would obtain measurements from the tree, the traffic-light and the taxi from the opposite lane. In addition to association, these predicted measurements also have values (e.g. pixels, distance) which depend on the state estimation (of both robot position and landmarks). Under an MPC framework, Figure 4.1.2b presents the succeeding inference for current time $k + 1$, in which our automated car advanced a bit more than planned, and indeed obtained three measurements. Two of these measurements are to the tree and the traffic-light (i.e. with consistent DA), while the third is to the couple that left the taxi (i.e. inconsistent DA). In such a case, merely updating the measurement values will not resolve the difference between the aforementioned DAs; instead the DA should be updated to match the acquired data, before updating the measurement values. We provide a novel paradigm to update inconsistent DA, leveraging iSAM2 graphical model based methodologies, thus setting the conditions for complete inference update via BSP regardless of DA consistency.

To summarize, our contributions in this chapter are as follows: (a) We introduce RUB Inference, a novel approach for saving computation time during the inference stage by reusing calculations made during the precursory planning stage; (b) We provide four exact methods, that utilize our concept under the assumption of consistent DA. We evaluate these four methods and compare them to the state of the art in simulation. (c) We provide a paradigm for incrementally updating inconsistent DA, thereby relaxing the afore-mentioned assumption; (d) We evaluate our complete paradigm and compare it to the state of the art both in simulation and on real-world data, considering the problem of autonomous navigation in unknown environments.

4.2 INFERENCE & BSP TODAY

In this work, we consider the joint inference and belief space planning problem in a model predictive control (MPC) setting, i.e. BSP is performed after each inference phase. This problem can be roughly divided into two successive and recursive stages, namely inference and planning. The former performs inference given all information up to current time, updating the belief over the state with incoming information (e.g. sensor measurements). The latter produces the next control action(s), given the belief from the former inference stage and a user defined objective function.

Let x_t denote the robot's state at time instant t and \mathcal{L} represent the world state if the latter is uncertain or unknown. For example, for SLAM problem, it could represent objects or 3D landmarks. The joint state, up to time k , is defined as

$$X_k = \{x_0, \dots, x_k, \mathcal{L}\} \in \mathbb{R}^n. \quad (4.1)$$

We shall be using the notation $t|k$ to refer to some time instant t while considering information up to time k ; as will be shown in the sequel, this notation will allow to refer to *sequential* inference and planning phases in a unified manner.

Let $z_{t|k}$ and $u_{t|k}$ denote, respectively, the measurements and the applied control action at time t , while the current time is k . For example, $z_{k+1|k}$ represents measurements from a future time instant $k+1$ while $z_{k-1|k}$ represents measurements from a past time instant $k-1$, with the present time being k in both cases. Representing the measurements and controls up to time t , given current time k , as

$$z_{1:t|k} \doteq \{z_{1|k}, \dots, z_{t|k}\}, \quad u_{0:t-1|k} \doteq \{u_{0|k}, \dots, u_{t-1|k}\}, \quad (4.2)$$

the posterior probability density function (pdf) over the joint state, denoted as the *belief*, is given by

$$b[X_t|k] \doteq \mathbb{P}(X_t|z_{1:t|k}, u_{0:t-1|k}). \quad (4.3)$$

For $t = k$, Eq. (4.3) represents the posterior at current time k , while for $t > k$ it represents planning stage posterior for a specific sequence of future actions and observations. Using Bayes rule, Eq. (4.3) can be rewritten as

$$\mathbb{P}(X_t|z_{1:t|k}, u_{0:t-1|k}) \propto \mathbb{P}(x_o) \cdot \prod_{i=1}^t \left[\mathbb{P}(x_i|x_{i-1}, u_{i-1|k}) \prod_{j \in \mathcal{M}_{i|k}} \mathbb{P}(z_{i|k}^j|x_i, l_j) \right], \quad (4.4)$$

where $\mathbb{P}(x_o)$ is the prior on the initial joint state, $\mathbb{P}(x_i|x_{i-1}, u_{i-1|k})$ and $\mathbb{P}(z_{i|k}^j|x_i, l_j)$ denote, respectively, the motion and measurement likelihood models. The set $\mathcal{M}_{i|k}$ contains all landmark indices observed at time i , i.e. it denotes data association (DA). The measurement of some landmark j at time i is denoted by $z_{i|k}^j \in z_{i|k}$. Under graphical representation of the belief, the conditional probabilities of the motion and observation models as well as the prior, can be denoted as factors (see Appendix-B). Eq. (4.4) can also be represented by a multiplication of these factors

$$\mathbb{P}(X_t|z_{1:t|k}, u_{0:t-1|k}) \propto \prod_{i=0}^t \{f_j\}_{i|k}, \quad (4.5)$$

where $\{f_j\}_{i|k}$ represents all factors added at time i while current time is k . The motion and measurement models are conventionally modeled with additive zero-mean Gaussian noise

$$x_{i+1} = f(x_i, u_i) + w_i, \quad w_i \sim \mathcal{N}(0, \Sigma_w) \quad (4.6)$$

$$z_i^j = h(x_i, l_j) + v_i, \quad v_i \sim \mathcal{N}(0, \Sigma_v), \quad (4.7)$$

where f and h are known possibly non-linear functions, Σ_w and Σ_v are the process and measurement noise covariance matrices respectively.

4.2.1 INFERENCE

For the inference problem, $t \leq k$, i.e time instances that are equal or smaller than current time. The maximum a posteriori (MAP) estimate of the joint state X_k for time $t = k$ is given by

$$X_{k|k}^* = \arg \max_{X_k} b[X_{k|k}] = \arg \max_{X_k} \mathbb{P}(X_k | z_{1:k|k}, u_{0:k-1|k}). \quad (4.8)$$

For the Gaussian case, the MAP solution produces the first two moments of the belief through solving a Non-linear Least Squares (NLS) problem, as will be shown later on. The MAP estimate from Eq. (4.8) is referred to as the *inference solution* in which, all controls and observations until time instant k are known.

4.2.2 PLANNING IN THE BELIEF SPACE

As mentioned, the purpose of planning is to determine the next optimal action(s). Finite horizon belief space planning for L look ahead steps involves inference over the beliefs

$$b[X_{k+l|k}] = \mathbb{P}(X_{k+l} | z_{1:k+l|k}, u_{0:k+l-1|k}) , \quad l \in [k+1, k+L] \quad (4.9)$$

where we use the same notation as in Eq. (4.3) to denote the current time is k . The belief (4.9) can be written recursively as a function of the belief $b[X_{k|k}]$ from the inference phase as

$$b[X_{k+l|k}] = b[X_{k|k}] \cdot \prod_{i=k+1}^{k+l} \left[\mathbb{P}(x_i | x_{i-1}, u_{i-1|k}) \prod_{j \in \mathcal{M}_{i|k}} \mathbb{P}(z_{i|k}^j | x_i, l_j) \right], \quad (4.10)$$

for the considered action sequence $u_{k:k+l-1|k}$ at planning time k , and observations $z_{k+1:k+l|k}$ that are expected to be obtained upon execution of these actions. The set $\mathcal{M}_{i|k}$ denotes landmark indices that are expected to be observed at a future time instant i . It is worth stressing that the future belief (4.10) is determined by a specific realization of unknown future observations $z_{k+1:k+l|k}$, as stated in the belief definition in (4.9). Since terms for future belief of the form $\mathbb{P}(X_{k+l} | z_{1:k+l|k}, u_{0:k+l-1|k})$ will be used frequently in this paper in order not to burden the reader we use the more compact form $b[X_{i|k}]$. Whenever $i > k$ the reader should consider the belief $b[X_{i|k}]$ as a function of a specific realization of future observations.

One can now define a general objective function

$$J(u_{k:k+L-1|k}) \doteq \mathbb{E}_{z_{k+1:k+L|k}} \left[\sum_{i=k+1}^{k+L} c_i (b[X_{i|k}], u_{i-1|k}) \right], \quad (4.11)$$

with immediate costs (or rewards) c_i and where the expectation considers all the possible realizations of the future observations $z_{k+1:k+L|k}$. Conceptually, one could also reason whether these observations will actually be obtained, e.g. by considering also different realizations of $\mathcal{M}_{i|k}$. Note that for Gaussian distributions considered herein and information-theoretic costs (e.g. entropy), it can be shown that the expectation operator can be omitted under maximum-likelihood observations assumption [29], while another alternative is to simulate future observations via sampling, e.g. [18, Section II-B], if such a simulator is available. The optimal open-loop control can now be defined as

$$u_{k:k+L-1|k}^* = \arg \min_{u_{k:k+L-1|k}} J(u_{k:k+L-1|k}). \quad (4.12)$$

Evaluating the objective function (4.11) for a candidate action sequence involves calculating belief evolution for the latter, i.e. solving the inference problem for each candidate action using predicted future associations and measurements. Note that since we consider an MPC framework, the optimal control is effectively not an open-loop control, since it is being recalculated at each single action step.

4.3 UTILIZING INFERENCE AND BSP SIMILARITIES

Calculating the next optimal action $u_{k|k}^* \in u_{k:k+L-1|k}^*$ within BSP necessarily involves inference over the belief $b[X_{k+1|k}]$ conditioned on the same action $u_{k|k}^*$. As we discuss in the sequel, this belief $b[X_{k+1|k}]$ can be different than $b[X_{k+1|k+1}]$ (the posterior at current time $k + 1$) due to partially inconsistent data association and difference between measurement values considered in planning and those obtained in practice in inference. Our approach for RUB Inference, takes care of both of these aspects, thereby enabling to obtain $b[X_{k+1|k+1}]$ from $b[X_{k+1|k}]$.

In the following, we analyze the similarities between inference and BSP and use them to formulate RUB Inference. It is worth stressing that the only thing needed to be changed in any BSP algorithm in order to support our paradigm for RUB Inference, is just adding more information to its output. More specifically, outputting not only the (sub)optimal action $u_{k|k}^*$, but also the corresponding future belief $b[X_{k+1|k}]$ (e.g. the difference between Figures 4.1.1a and 4.1.1b).

4.3.1 LOOKING INTO INFERENCE

To better understand the similarities between inference and precursory planning, let us break down the inference solution to its components. Introducing Eqs. (4.4-4.7) into Eq. (4.8) and taking the negative logarithm yields the following non-linear least squares problem (NLS)

$$X_{k|k}^* = \arg \min_{X_k} \|x_o - x_o^*\|_{\Sigma_o}^2 + \sum_{i=1}^k \left[\|x_i - f(x_{i-1}, u_{i-1|k})\|_{\Sigma_w}^2 + \sum_{j \in \mathcal{M}_i|k} \|z_{i|k}^j - h(x_i, l_j)\|_{\Sigma_v}^2 \right], \quad (4.13)$$

where $\|a\|_{\Sigma}^2 \doteq a^T \Sigma^{-1} a$ is the squared Mahalanobis norm.

Linearizing each of the terms in Eq. (4.13) and performing standard algebraic manipulations (see Appendix B for derivation) yields

$$\Delta X_{k|k}^* = \arg \min_{\Delta X_k} \|A_{k|k} \Delta X_k - b_{k|k}\|^2, \quad (4.14)$$

where $A_{k|k} \in \mathbb{R}^{m \times n}$ is the Jacobian matrix and $b_{k|k} \in \mathbb{R}^m$ is the right hand side (RHS) vector. In a more elaborated representation

$$A_{k|k} = \begin{bmatrix} \Sigma_o^{-\frac{1}{2}} \\ \mathcal{F}_{1:k|k} \\ \mathcal{H}_{1:k|k} \end{bmatrix}, \quad b_{k|k} = \begin{bmatrix} o \\ \check{b}_{1:k|k}^{\mathcal{F}} \\ \check{b}_{1:k|k}^{\mathcal{H}} \end{bmatrix}, \quad (4.15)$$

where $\mathcal{F}_{1:k|k}$, $\mathcal{H}_{1:k|k}$, $\check{b}_{1:k|k}^{\mathcal{F}}$ and $\check{b}_{1:k|k}^{\mathcal{H}}$ (see Appendix A) denote the Jacobian matrices and RHS vectors of all motion and observation terms accordingly, for time instances $1 : k$ when the current time is k . These Jacobians, along with the corresponding RHS can be referred to by

$$\mathcal{A}_{1:k|k} = \begin{bmatrix} \mathcal{F}_{1:k|k} \\ \mathcal{H}_{1:k|k} \end{bmatrix}, \quad \check{b}_{1:k|k} = \begin{bmatrix} \check{b}_{1:k|k}^{\mathcal{F}} \\ \check{b}_{1:k|k}^{\mathcal{H}} \end{bmatrix}, \quad (4.16)$$

While there are a few methods to solve Eq. (4.14), we choose QR factorization as presented, e.g., in [33]. The QR factorization of the Jacobian matrix $A_{k|k}$ is given by the orthonormal rotation matrix $Q_{k|k}$ and the upper triangular matrix $R_{k|k}$

$$A_{k|k} = Q_{k|k} R_{k|k}. \quad (4.17)$$

Eq. (4.17) is introduced into Eq. (4.14), thus producing

$$R_{k|k} \Delta X_k = d_{k|k}, \quad (4.18)$$

where $R_{k|k}$ is an upper triangular matrix and $d_{k|k}$ is the corresponding RHS vector, given by the original RHS vector and the orthonormal rotation matrix $Q_{k|k}$

$$d_{k|k} \doteq Q_{k|k}^T b_{k|k}. \quad (4.19)$$

We can now solve Eq. (4.18) for ΔX_k via back substitution, update the linearization point, and repeat the process until convergence. Eq. (4.18) can also be presented using a Bayes tree (BT) [34]. A BT is a graphical representation of a factorized Jacobian matrix (the square root information matrix) R and the corresponding RHS vector d , in the form of a directed tree. More on the formulation of inference using graphical models can be found in Appendix B. One can substantially reduce running time by exploiting sparsity and updating the QR factorization from the previous step with new information instead of calculating a factorization from scratch, see e.g. iSAM2 algorithm [36].

Given the inference solution, the belief $b[X_{k|k}]$ can be approximated by the Gaussian

$$b[X_{k|k}] \doteq \mathbb{P}(X_k | z_{1:k|k}, u_{o:k-1|k}) = \mathcal{N}(X_{k|k}^*, \Lambda_{k|k}^{-1}), \quad (4.20)$$

while the information matrix is given by

$$\Lambda_{k|k} = A_{k|k}^T A_{k|k} = R_{k|k}^T R_{k|k}, \quad (4.21)$$

and the factorized Jacobian matrix $R_{k|k}$ along with the corresponding RHS vector $d_{k|k}$ can be used to update the linearization point and to recover the MAP estimate. In other words, the factorized Jacobian matrix $R_{k|k}$ and the corresponding RHS vector $d_{k|k}$ are sufficient for performing a single iteration within Gaussian belief inference.

4.3.2 LOOKING INTO PLANNING

An interesting insight, that will be exploited in the sequel, is that the underlying equations of BSP are similar to those seen in Section 4.3.1. In particular, evaluating the belief at the L th look ahead step, $b[X_{k+L|k}]$, involves MAP inference over a certain action sequence $u_{k:k+l-1|k}$ and future measurements

$z_{k+1:k+L|k}$, which in turn, as in Section 4.3.1, can be described as an NLS problem

$$X_{k+L|k}^* = \arg \min_{X_{k+L}} \|X_k - X_{k|k}^*\|_{\Lambda_{k|k}^{-1}}^2 + \sum_{i=k+1}^{k+L} \left[\|x_i - f(x_{i-1}, u_{i-1|k})\|_{\Sigma_w}^2 + \sum_{j \in \mathcal{M}_{i|k}} \|z_{i|k}^j - h(x_i, l_j)\|_{\Sigma_v}^2 \right] \quad (4.22)$$

For $i > k$, the set $\mathcal{M}_{i|k}$ contains *predicted* associations for future time instant i ; hence, we can claim that $\forall i > k$ it is possible that $\mathcal{M}_{i|k} \neq \mathcal{M}_{i|i}$. In other words, it is possible that associations from the planning stage, $\mathcal{M}_{k+1|k}$, would be partially different than the associations from the corresponding inference stage $\mathcal{M}_{k+1|k+1}$. Moreover, the likelihood for inconsistent DA between planning and the corresponding inference rises as we look further into the future, i.e. with the distance $\|i - k\|$ increasing; e.g. $\mathcal{M}_{k+j|k}$ and $\mathcal{M}_{k+j|k+j}$ are less likely to be identical for $j = 10$ than they are for $j = 1$.

Predicting the unknown measurements $z_{k+1:k+L|k}$ in terms of both association and values can be done in various ways. In this paper the DA is predicted using current state estimation, and measurement values are obtained using the maximum-likelihood (ML) assumption, i.e. assuming zero innovation [14]. The robot pose is first propagated using the motion model (4.6). All landmark estimations are then transformed to the robot's new camera frame. Once in the robot camera frame, all landmarks that are within the robot's field of view are considered to be seen by the robot (predicted DA). The estimated position of each landmark, that is considered as visible by the robot, is being projected to the camera image plane [26], thus generating measurements. It is worth mentioning that the aforementioned methodology is not able to predict occurrences of new landmarks, since it is based solely on the map the robot built thus far, i.e. current joint state estimation. The ability to predict occurrences of new landmarks would increase the advantage of RUB Inference over conventional Bayesian inference (as discussed in the sequel), hence is left for future work.

Once the predicted measurements are acquired, by following a similar procedure to the one presented in Section 4.3.1, for each action sequence we get

$$\Delta X_{k+L|k}^* = \arg \min_{\Delta X_{k+L}} \|A_{k+L|k} \Delta X_{k+L} - b_{k+L|k}\|^2. \quad (4.23)$$

The Jacobian matrix $A_{k+L|k}$ and RHS vector $b_{k+L|k}$ are defined as

$$A_{k+L|k} \doteq \begin{bmatrix} A_{k|k} \\ \mathcal{A}_{k+1:k+L|k} \end{bmatrix}, \quad b_{k+L|k} \doteq \begin{bmatrix} b_{k|k} \\ \check{b}_{k+1:k+L|k} \end{bmatrix}, \quad (4.24)$$

where $A_{k|k}$ and $b_{k|k}$ are taken from inference, see Eq. (4.14), and $\mathcal{A}_{k+1:k+L|k}$ and $\check{b}_{k+1:k+L|k}$ correspond to the new terms obtained at the first L look ahead steps (e.g. see Eq. (4.16)). Note that although

$\mathcal{A}_{k+1:k+L|k}$ is not a function of the (unknown) measurements $z_{k+1:k+L|k}$, it is a function of the predicted DA, $\mathcal{M}_{k+1:k+L|k}$ [29]. Performing QR factorization, yields

$$A_{k+L|k} = Q_{k+L|k}^A R_{k+L|k}, \quad (4.25)$$

from which the information matrix, required in the information-theoretic cost, can be calculated. Using Eq. (4.24) the belief that correlates to the specific action sequence can be estimated, enabling evaluating the objective function (4.11). Determining the best action via Eq. (4.12) involves repeating this process for different candidate actions.

4.3.3 SIMILARITIES BETWEEN INFERENCE AND BSP

In an MPC setting, only the first action from the sequence $u_{k:k+L-1|k}^*$ is executed, i.e.

$$u_{k|k+1} = u_{k|k}^* \in u_{k:k+L-1|k}^*. \quad (4.26)$$

In such case the difference between the belief obtained from BSP (for action $u_{k|k}^*$)

$$b[X_{k+1|k}] \equiv \mathbb{P}(X_{k+1}|z_{1:k|k}, u_{0:k-1|k}, z_{k+1|k}, u_{k|k}^*), \quad (4.27)$$

and the belief from the succeeding inference

$$b[X_{k+1|k+1}] \equiv \mathbb{P}(X_{k+1}|z_{1:k|k}, u_{0:k-1|k}, z_{k+1|k+1}, u_{k|k+1}), \quad (4.28)$$

is rooted in the set of measurements (i.e. $z_{k+1|k+1}$ vs. $z_{k+1|k}$), and the corresponding factors added at time instant $k + 1$. These factor sets, denoted by $\{f_i\}_{k+1|k}$ and $\{f_j\}_{k+1|k+1}$ accordingly, can differ from one another in data association and measurement values. Since solving the belief requires linearization (4.14), it is important to note that both beliefs, $b[X_{k+1|k}]$ and $b[X_{k+1|k+1}]$, make use of the *same* initial linearization point \bar{X}_{k+1} for the common variables. In particular, as in this work we do not reason within planning about new, unmapped thus far, landmarks, it follows that

$$X_{k+1|k} = \begin{bmatrix} X_{k|k} \\ x_{k+1} \end{bmatrix}, \quad X_{k+1|k+1} = \begin{bmatrix} X_{k|k} \\ x_{k+1} \\ L_{k+1}^{new} \end{bmatrix} \quad (4.29)$$

where L_{k+1}^{new} represents the new landmarks that were added to the belief for the first time at time instant $k + 1$. The linearization point for the common variables is $[X_{k|k}^*, f(x_k, u_{k|k}^*)]$ for planning,

and $[X_{k|k}^*, f(x_k, u_{k|k+1})]$ for succeeding inference, where $f(\cdot)$ is the motion model (4.6). Since the (sub)optimal action provided by BSP is the one executed in the succeeding inference i.e. Eq. (4.26), the motion models are identical hence the same linearization point is used in both inference and precursory planning.

When considering the belief from planning (4.27), which is propagated with the next action (4.26) and predicted measurements, with the previously factorized form of $A_{k|k}$ and $b_{k|k}$, we get

$$A_{k+1|k}^R \doteq \begin{bmatrix} R_{k|k} \\ \mathcal{A}_{k+1|k} \end{bmatrix}, \quad b_{k+1|k}^d \doteq \begin{bmatrix} d_{k|k} \\ \check{b}_{k+1|k} \end{bmatrix}. \quad (4.30)$$

Similarly, when considering the a posteriori belief from inference (4.28), propagated with the next action (4.26) and acquired measurements, with the previously factorized form of $A_{k|k}$ and $b_{k|k}$, we get

$$A_{k+1|k_1}^R \doteq \begin{bmatrix} R_{k|k} \\ \mathcal{A}_{k+1|k+1} \end{bmatrix}, \quad b_{k+1|k+1}^d \doteq \begin{bmatrix} d_{k|k} \\ \check{b}_{k+1|k+1} \end{bmatrix}. \quad (4.31)$$

For the same action (4.26), the difference between Eq. (4.30) to the equivalent representation of standard Bayesian inference (4.31) originates from the factors added at time $k + 1$

$$\mathcal{A}_{k+1|k} \stackrel{?}{=} \mathcal{A}_{k+1|k+1}, \quad (4.32)$$

$$\check{b}_{k+1|k} \stackrel{?}{=} \check{b}_{k+1|k+1}. \quad (4.33)$$

Since the aforementioned share the same action sequence, the same linearization point and the same models, the differences remain limited to the DA and measurement values at time $k + 1$.

In planning, DA is based on predicting which landmarks would be observed. This DA could very possibly be different than the actual landmarks the robot observes, as presented in Sec. 4.3.2. This inconsistency in DA manifests in both the Jacobian matrices and the RHS vectors. Even in case of consistent DA, the predicted measurements (if exist) would still be different than the actual measurements due to various reasons, e.g. the predicted position is different than the ground truth of the robot, measurement noise, inaccurate models.

While for consistent DA and the same linearization point Eq. (4.32) will always be true, the RHS vectors, specifically Eq. (4.33), would still be different due to the difference in measurement values considered in planning and actually obtained in inference.

It is worth stressing that consistent data association between inference and precursory planning suggests that all predictions for state variable (new or existing) associations were in fact true. In addi-

tion to the new robot state added each time instant, new variables could also manifest in the form of landmarks. Consistent DA implies that the future appearance of all new landmarks has been perfectly predicted during planning. Since for the purpose of this work, we use a simple prediction mechanism unable to predict new landmarks (see Section 4.3.2), consistent DA would inevitably mean no new landmarks in inference, i.e. L_{k+1}^{new} is an empty set.

We start developing RUB Inference by assuming consistent DA between inference and precursory planning (Section 4.4). In such a case the difference is limited to the RHS vectors. Later we relax this assumption by dealing with possible DA inconsistency prior to the update of the RHS vector, thus addressing the general and complete problem of inference update using RUB Inference paradigm (Section 4.6).

4.4 CONSISTENT DA ASSUMPTION

Let us assume that the DA between inference and precursory planning is consistent, whether the cause is a "lucky guess" during planning or whether the DA inconsistency has been resolved beforehand. Recalling the definition of $\mathcal{M}_{i|k}$ (see e.g. Eq. (4.10)), this assumption is equivalent to writing

$$\mathcal{M}_{k+1|k} \equiv \mathcal{M}_{k+1|k+1}. \quad (4.34)$$

In other words, landmarks considered to be observed at a future time $k + 1$, will indeed be observed at that time. Note this does *not* necessarily imply that actual measurements and robot poses will be as considered within the planning stage, but it does necessarily state that both are considering the same variables and the same associations.

We now observe that the motion models in both $b[X_{k+1|k+1}]$ and $b[X_{k+1|k}]$ are evaluated considering the *same* control (i.e. the optimal control u_k^*). Moreover, the robot pose x_{k+1} is initialized to the *same* value in both cases as $f(x_k, u_k^*)$, see e.g. Eq.(27) in [29], and thus the linearization point of all probabilistic terms in inference and planning is *identical*. This, together with the aforementioned assumption (i.e. Eq. (4.34) holds) allows us to write $A_{k+1|k} = A_{k+1|k+1}$, and hence

$$R_{k+1|k+1} \equiv R_{k+1|k}, \quad (4.35)$$

for the *first iteration* in the inference stage at time $k + 1$.

Hence, in order to solve $b[X_{k+1|k+1}]$ we are left to find the RHS vector $d_{k+1|k+1}$, while $R_{k+1|k+1}$ can be *entirely re-used*.

In the sequel we present four methods that can be used for updating the RHS vector, and examine

Variable	Description
$\square_{t k}$	Of time t while current time is k
ΔX_k	State perturbation around linearization point
$\mathcal{M}_{t k}$	Data Association at time t
$A_{t k}$	Jacobian matrix at time t
$b_{t k}$	RHS vector at time t
$\mathcal{A}_{t k}$	Jacobian part related to all factors added at time t
$\mathcal{F}_{t k}$	Jacobian part related to motion factor added at time t
$\mathcal{H}_{t k}$	Jacobian part related to all factors added at time t without the motion factor
$\check{b}_{t k}$	RHS vector related to all factors added at time t
$\check{b}_{t k}^{\mathcal{F}}$	RHS vector related to motion factor at time t
$\check{b}_{t k}^{\mathcal{H}}$	RHS vector related to all factors added at time t without the motion factor
$R_{t k}$	Factorized Jacobian, i.e. square root information matrix
$d_{t k}$	Factorized RHS vector
$A_{t k}^R$	Factorized $\left[R_{t-1 k}^T, \mathcal{A}_{t k}^T \right]^T$
$R_{t k}^{\mathcal{F}}$	Factorized $\left[R_{t-1 k}^T, \mathcal{F}_{t k}^T \right]^T$
$d_{t k}^{\mathcal{F}}$	Factorized $\left[d_{t-1 k}^T, \check{b}_{t k}^{\mathcal{F}T} \right]^T$
$R_{t k}^{aug}$	Factorized Jacobian at time $t - 1$ zero padded to match factorized Jacobian at time t
$d_{t k}^{aug}$	Factorized RHS vector at time $t - 1$ zero padded to match factorized RHS vector at time t
$Q_{t k}^A$	Rotation matrix for factorizing $A_{t k}$ into $R_{t k}$
$Q_{t k}$	Rotation matrix for factorizing $A_{t k}^R$ into $R_{t k}$
$Q_{t k}^{\mathcal{F}}$	Rotation matrix for factorizing $\left[R_{t-1 k}^T, \mathcal{F}_{t k}^T \right]^T$ into $R_{t k}^{\mathcal{F}}$
$Q_{t k}^{\mathcal{H}}$	Rotation matrix for factorizing $\left[R_{t k}^{\mathcal{F}T}, \mathcal{H}_{t k}^T \right]^T$ into $R_{t k}$

Table 4.1: Notations for Section 4.4

computational aspects of each. The four methods use two different approaches to update the RHS vector: while the first two (OTM and OTM-OO), utilize the rotation matrix available from factorization, the last two (DU and DU-OO) utilize information downdate / update principles. After we review the methods we shortly discuss the advantages and disadvantages of each (Sec. 4.4.5). It is worth stressing that each of these methods results in the same RHS vector which is also identical to the RHS vector that would have been obtained by the standard inference update. With both the factorized Jacobian matrix (i.e. R) and the RHS vector identical to the standard inference update approach, RUB inference provides the same estimation accuracy for the inference solution.

4.4.1 THE ORTHOGONAL TRANSFORMATION MATRIX METHOD - OTM

In the OTM method, we obtain $d_{k+1|k+1}$ following the definition as written in Eq. (4.19). Recall that at time $k + 1$ in the inference stage, the posterior should be updated with new terms that correspond, for example, to motion model and obtained measurements. The RHS vector's augmentation, that corresponds to these new terms is denoted by $\check{b}_{k+1|k+1}$, see Eq. (4.16). Given $R_{k|k}$ and $d_{k|k}$ from the inference stage at time k , the augmented system at time $k + 1$ is

$$A_{k+1|k+1}^R \Delta X_{k+1} \doteq \begin{bmatrix} R_{k|k} \\ \mathcal{A}_{k+1|k+1} \end{bmatrix} \Delta X_{k+1} = \begin{bmatrix} d_{k|k} \\ \check{b}_{k+1|k+1} \end{bmatrix} \quad (4.36)$$

which after factorization of $A_{k+1|k+1}^R$ (see Eqs. (4.17)-(4.19)) becomes

$$R_{k+1|k+1} \Delta X_{k+1} = d_{k+1|k+1}, \quad (4.37)$$

where

$$d_{k+1|k+1} = Q_{k+1|k+1}^T \begin{bmatrix} d_{k|k} \\ \check{b}_{k+1|k+1} \end{bmatrix}. \quad (4.38)$$

As deduced from Eq. (4.38), the calculation of $d_{k+1|k+1}$ requires $Q_{k+1|k+1}$. Since $A_{k+1|k}^R \equiv A_{k+1|k+1}^R$ (see Section 4.4), we get $Q_{k+1|k+1} = Q_{k+1|k}$. However, $Q_{k+1|k}$ is already available from the precursory planning stage, see Eq. (4.25), and thus calculating $d_{k+1|k+1}$ via Eq. (4.38) does *not* involve QR factorization in practice. To summarize, under the OTM method we obtain the RHS vector $d_{k+1|k+1}$ in the following manner:

$$d_{k+1|k+1} = Q_{k+1|k}^T \begin{bmatrix} d_{k|k} \\ \check{b}_{k+1|k+1} \end{bmatrix}. \quad (4.39)$$

where $Q_{k+1|k}^T$ is available from the factorization of precursory planning, $d_{k|k}$ is the RHS from inference at time k , and $\check{b}_{k+1|k+1}$ are the new un-factorized RHS values obtained at time $k + 1$.

4.4.2 THE OTM - ONLY OBSERVATIONS METHOD - OTM-OO

The OTM-OO method is a variant of the OTM method. OTM-OO aspires to utilize even more information from the planning stage. Since the motion models from inference and the precursory planning first step are identical, i.e. same function $f(., .)$, see Eqs. (4.13) and (4.22), and as in both cases the *same* control is considered - Eq. (4.26), there is no reason to change the motion model data from the RHS vector $d_{k+1|k}$. In order to enable the aforementioned, we require the matching rotation matrix. One way would be to break down the planning stage as described in Section 4.3.2 into two stages, in which the motion and observation models are updated separately. Usually this breakdown is performed either way since a propagated future pose is required for predicting future measurements.

So following Section 4.4.1, instead of using $d_{k|k}$, we attain from planning the RHS vector already with the motion model ($d_{k+1|k}^{\mathcal{F}}$), augment it with the new measurements and rotate it with the corresponding rotation matrix obtained from the planning stage

$$d_{k+1|k+1} = Q_{k+1|k}^{\mathcal{H}^T} \begin{bmatrix} d_{k+1|k}^{\mathcal{F}} \\ \check{b}_{k+1|k+1}^{\mathcal{H}} \end{bmatrix}. \quad (4.40)$$

The rotation matrix $Q_{k+1|k}^{\mathcal{H}}$ is given from the precursory planning stage where

$$Q_{k+1|k}^{\mathcal{H}} R_{k+1|k} = \begin{bmatrix} R_{k+1|k}^{\mathcal{F}} \\ \mathcal{H}_{k+1|k} \end{bmatrix}, \quad (4.41)$$

and where $R_{k+1|k}^{\mathcal{F}}$ is the factorized Jacobian propagated with the motion model given by

$$Q_{k+1|k}^{\mathcal{F}} R_{k+1|k}^{\mathcal{F}} = \begin{bmatrix} R_{k|k} \\ \mathcal{F}_{k+1|k} \end{bmatrix}. \quad (4.42)$$

As will be seen later on, the OTM-OO method would prove to be the most computationally efficient between the four suggested methods.

4.4.3 THE DOWNDATE UPDATE METHOD - DU

In the DU method we propose to re-use the $d_{k+1|k}$ vector from the planning stage to calculate $d_{k+1|k+1}$.

While not necessarily required within the planning stage, $d_{k+1|k}$ could be calculated at that stage from $b_{k+1|k}$ and $Q_{k+1|k}$, see Eqs. (4.24)-(4.25). However, $b_{k+1|k}$ (unlike $A_{k+1|k}$) is a function of the unknown future observations $z_{k+1|k}$, which would seem to complicate things. Our solution to this issue is as follows: We assume *some* value for the observations $z_{k+1|k}$ and then calculate $d_{k+1|k}$ within

the planning stage. As in inference at time $k + 1$, the actual measurements $z_{k+1|k+1}$ will be different, we remove the contribution of $z_{k+1|k}$ to $d_{k+1|k}$ via information downdating [11, Sec. V-A], and then appropriately incorporate $z_{k+1|k+1}$ to get $d_{k+1|k+1}$ using the same mechanism.

More specifically, downdating the measurements $z_{k+1|k}$ from $d_{k+1|k}$ is done via [11, Sec. V-A]

$$d_{k+1|k}^{aug} = R_{k+1|k}^{aug^{-T}} (R_{k+1|k}^T d_{k+1|k} - \mathcal{A}_{k+1|k}^T \check{b}_{k+1|k}), \quad (4.43)$$

where $\check{b}_{k+1|k}$ is a function of $z_{k+1|k}$, see Eqs. (4.22)-(4.24), and where $R_{k+1|k}^{aug}$ is the downdated $R_{k+1|k}$ matrix which is given by

$$R_{k+1|k}^{aug^T} R_{k+1|k}^{aug} = A_{k+1|k}^R A_{k+1|k}^R - \mathcal{A}_{k+1|k}^T \mathcal{A}_{k+1|k}. \quad (4.44)$$

Interestingly, the above calculations are not really required: Since we already have $d_{k|k}$ from the previous inference stage, we can attain the downdated $d_{k+1|k}^{aug}$ vector more efficiently by augmenting $d_{k|k}$ with zero padding.

$$d_{k+1|k}^{aug} = \begin{bmatrix} d_{k|k} \\ \text{o} \end{bmatrix} \quad (4.45)$$

where $d_{k+1|k}^{aug}$ is the downdated RHS vector and o is a zero padding to match dimensions. Similarly, $R_{k+1|k}^{aug}$ can be calculated as

$$R_{k+1|k}^{aug} = \begin{bmatrix} R_{k|k} & \text{o} \\ \text{o} & \text{o} \end{bmatrix}, \quad (4.46)$$

where $R_{k|k}$ is zero padded to match dimensions of $R_{k+1|k}$.

Now, all which is left to get $d_{k+1|k+1}$, is to incorporate the new measurements $z_{k+1|k+1}$ (encoded in $\check{b}_{k+1|k+1}$). We utilize the information downdating mechanism in [11, Sec. V-A], in order to update information. Intuitively, instead of downdating information from $d_{k+1|k}$, we would like to add information to $d_{k+1|k}^{aug}$. So by appropriately adjusting Eq.(4.43) this can be done via

$$d_{k+1|k+1} = R_{k+1|k+1}^{-T} (R_{k+1|k}^{aug^T} d_{k+1|k}^{aug} + \mathcal{A}_{k+1|k+1}^T \check{b}_{k+1|k+1}), \quad (4.47)$$

where according to Eq. (4.34) $R_{k+1|k+1} \equiv R_{k+1|k}$ and $\mathcal{A}_{k+1|k+1} \equiv \mathcal{A}_{k+1|k}$, $R_{k+1|k}^{aug}$ is given by Eq.(4.46), $d_{k+1|k}^{aug}$ is given by Eq.(4.45), and $\check{b}_{k+1|k+1}$ are the new un-factorized RHS values obtained at time $k + 1$.

To summarize, under the DU method we obtain the RHS vector $d_{k+1|k+1}$ in the following manner:

$$d_{k+1|k+1} = R_{k+1|k}^{-T} \left(\begin{bmatrix} R_{k|k} & \text{o} \\ \text{o} & \text{o} \end{bmatrix}^T \begin{bmatrix} d_{k|k} \\ \text{o} \end{bmatrix} + \mathcal{A}_{k+1|k}^T \check{b}_{k+1|k+1} \right). \quad (4.48)$$

4.4.4 THE DU - ONLY OBSERVATIONS METHOD - DU-OO

The DU-OO method is a variant of the DU method, where, similarly to Section 4.4.2, we utilize the fact that there is no reason to change the motion model data from the RHS vector $d_{k+1|k}$. Hence we would downdate all data with the exception of the motion model, and then update accordingly. As opposed to Section 4.4.3, now we do need to downdate using [11, Sec. V-A]

$$d_{k+1|k}^{\mathcal{F}} = R_{k+1|k}^{\mathcal{F}-T} (R_{k+1|k}^T d_{k+1|k} - \mathcal{H}_{k+1|k}^T \check{b}_{k+1|k}^{\mathcal{H}}), \quad (4.49)$$

where $d_{k+1|k}^{\mathcal{F}}$ is the RHS vector, downdated from all predicted measurements with the exception of the motion model, and $R_{k+1|k}^{\mathcal{F}}$ is the equivalent downdated $R_{k+1|k}$ matrix which is given by

$$R_{k+1|k}^{\mathcal{F}^T} R_{k+1|k}^{\mathcal{F}} = A_{k+1|k}^{R^T} A_{k+1|k}^R - \mathcal{H}_{k+1|k}^T \mathcal{H}_{k+1|k}, \quad (4.50)$$

where $\mathcal{H}_{k+1|k}$ denotes the portion of the planning stage Jacobian, of the predicted factors with the exception of the motion model. Now, all which is left, is to update $d_{k+1|k}^{\mathcal{F}}$ with the new measurements from the inference stage

$$d_{k+1|k+1} = R_{k+1|k+1}^{-T} (R_{k+1|k}^{\mathcal{F}^T} d_{k+1|k}^{\mathcal{F}} + \mathcal{H}_{k+1|k+1}^T \check{b}_{k+1|k+1}^{\mathcal{H}}), \quad (4.51)$$

where according to Eq. (4.34) $R_{k+1|k+1} \equiv R_{k+1|k}$ and $\mathcal{H}_{k+1|k+1} \equiv \mathcal{H}_{k+1|k}$, $R_{k+1|k}^{\mathcal{F}}$ is given by Eq.(4.50), $d_{k+1|k}^{\mathcal{F}}$ is given by Eq.(4.49), and $\check{b}_{k+1|k+1}$ are the new un-factorized RHS values obtained at time $k+1$.

By introducing Eq. (4.49) into Eq.(4.51) we can also avert from calculating $R_{k+1|k}^{\mathcal{F}}$ so under the DU-OO assumption we obtain the RHS vector $d_{k+1|k+1}$ in the following manner:

$$d_{k+1|k+1} = R_{k+1|k}^{-T} \left(R_{k+1|k}^T d_{k+1|k} + \mathcal{H}_{k+1|k}^T \left(\check{b}_{k+1|k+1}^{\mathcal{H}} - \check{b}_{k+1|k}^{\mathcal{H}} \right) \right), \quad (4.52)$$

which can be rewritten as

$$d_{k+1|k+1} = d_{k+1|k} + R_{k+1|k}^{-T} \mathcal{H}_{k+1|k}^T \left(\check{b}_{k+1|k+1}^{\mathcal{H}} - \check{b}_{k+1|k}^{\mathcal{H}} \right). \quad (4.53)$$

4.4.5 DISCUSSION - RHS UPDATE METHODS

In this section we would like to give the reader some intuition regarding the advantages and disadvantages of the OTM approach when compared to the DU approach. Since both provide the same desired solution, the difference between them would manifest in computation time and ease of use. In the sequel we cover both starting with the complexity of each.

Figure 4.4.1: Illustration of the Jacobian matrix $A_{k+1|k+1}^R$, introduced in Eq. (4.31), on its components and dimensions. These notations are used along Section 4.4.5, and brought here for the reader's convenience.

Let us compare the complexity required for updating the RHS by OTM, see Eq. (4.39), against the complexity required for updating the RHS by DU, see Eq. (4.48). For OTM we have a single multiplication between a sparse rotation matrix $Q_{k+1|k}$ and a vector, both in the dimension of the joint state at time k plus the number of rows of the linearized new factors (i.e. depending on number of factors and their types). The complexity of OTM would be given by the number of non zeros in the rotation matrix $Q_{k+1|k}$. In Appendix C we provide some understanding on the creation of the rotation matrix $Q_{k+1|k}$, and also develop an expression for the number of non zeros in $Q_{k+1|k}$. We direct the reader to Figure 4.4.1 for illustration of the new notations used in this discussion. Following the development in Appendix C, the number of non zeros in $Q_{k+1|k}$ is represented by two potentially dominant terms separated by a simple condition

$$O(\text{OTM}) = \begin{cases} O\left(\left(n_k^s + n_{k+1}^f - j\right)^2\right) & n_{k+1}^f \geq n_k^s \geq 6 \\ O\left(n_k^s \cdot n_{k+1}^f\right) & n_{k+1}^f < n_k^s \end{cases} \quad (4.54)$$

where j denotes the column index of the left-most entry in $\mathcal{A}_{k+1|k+1}$, n_k^s denotes the size of the joint state vector at the precursory time k , n_{k+1}^f denotes the number of rows in the linearized new factors $\mathcal{A}_{k+1|k+1}$. The condition in Eq. (4.54) is a simple upper bound to the real expression (see Eq. (C.20)), resulting with a cleaner condition without affecting the solution.

It is worth stressing that depending on its type, each state occupies more than a single row / column in the Jacobian, e.g. 6DOF robot pose occupies six rows and six columns. Similarly, depending on its type, each factor occupies more than a single row in the Jacobian, e.g. a monocular factor occupies two rows in the Jacobian.

For DU in addition to multiplications between upper triangular matrices and vectors, we have a matrix inverse. Differently from OTM here the matrix dimensions are of the joint state vector at time

$k + 1$, hence the worst case scenario for DU is a fully dense upper triangular matrix inverse

$$O(\text{DU}) = O(n_{k+1}^s), \quad (4.55)$$

where n_{k+1}^s represents the size of the joint state vector at time $k + 1$.

For the case of $n_{k+1}^f < n_k^s$ we should compare

$$n_k^s \cdot n_{k+1}^f \stackrel{?}{\leq} n_{k+1}^s \cdot n_{k+1}^s. \quad (4.56)$$

Assuming states are not removed from the state vector, we can say

$$n_k^s \leq n_{k+1}^s, \quad (4.57)$$

then evidently

$$n_k^s \cdot n_{k+1}^f < n_{k+1}^s \cdot n_{k+1}^s. \quad (4.58)$$

For the case of $n_{k+1}^f \geq n_k^s \geq 6$ we should compare

$$n_k^s + n_{k+1}^f - j \stackrel{?}{\leq} n_{k+1}^s, \quad , \quad j \in [1, n_k^s], \quad (4.59)$$

so for this case OTM is computationally superior to DU if

$$n_{k+1}^f < n_{k+1}^s - n_k^s + j, \quad , \quad j \in [1, n_k^s]. \quad (4.60)$$

It is worth stressing that unlike Eq. (4.58), Eq. (4.60) is dependent on state ordering in the form of the left-most non zero entry in $\mathcal{A}_{k+1|k+1}$.

Concluding the complexity analysis of OTM and DU, OTM will be computationally superior to DU if the following holds

$$\left(n_{k+1}^f < n_k^s \right) \cup \left(n_{k+1}^f < n_{k+1}^s - n_k^s + j \cap n_{k+1}^f \geq n_k^s \right). \quad (4.61)$$

In other words, if the number of rows in $\mathcal{A}_{k+1|k+1}$ is smaller than the size of the state vector at time k OTM is computationally superior to DU. If the number of rows in $\mathcal{A}_{k+1|k+1}$ is larger or equal to the size of the state vector at time k , than OTM is computationally superior to DU only if the number of rows in $\mathcal{A}_{k+1|k+1}$ is smaller than the size of the added states at time $k + 1$ plus the column index of the left-most state in $\mathcal{A}_{k+1|k+1}$.

Although most of the time DU is computationally inferior, unlike OTM that requires access to

the rotation matrix which might not be easily available in every planning paradigm, DU makes use in a more readily available information: the inference solution of precursory time, the predicted factors, the new RHS vector at time $k+1$, and the factorized Jacobian from precursory planning. Therefore the advantage in using DU lies in the information availability with minimal adjustments to the planning stage.

Since OTM-OO would prove to perform the best empirically, let us get some intuition on why it is more efficient than OTM. The OO addition to OTM, refers to the use of the motion propagated belief $R_{k+1|k}^F d_{k+1|k}^F$ rather than the use of precursory inference solution $R_{k|k} d_{k|k}$. The dimension of $R_{k+1|k}^F$ is larger from that of $R_{k|k}$ by a single robot pose, while the number of rows of $\mathcal{H}_{k+1|k+1}$ is smaller by a single robot pose from that of $\mathcal{A}_{k+1|k+1}$. Let us assume without affecting generality that our robot pose dimension is a . Under this assumption we can calculate Eq. (4.54) for both OTM and OTM-OO. Let n_{k+1}^{-f} denote the number of rows of the newly added factors at time $k+1$ without the motion factor, i.e. $\mathcal{H}_{k+1|k+1}$ number of rows, so the complexity of OTM would be

$$O(\text{OTM}) = \begin{cases} O\left(\left(n_k^s + (n_{k+1}^{-f} + a) - j\right)^2\right) & (n_{k+1}^{-f} + a) \geq n_k^s \\ O\left(n_k^s \cdot (n_{k+1}^{-f} + a)\right) = O\left(n_k^s \cdot n_{k+1}^{-f} + a \cdot n_k^s\right) & (n_{k+1}^{-f} + a) < n_k^s \end{cases}, \quad (4.62)$$

while the complexity of OTM-OO would be

$$O(\text{OTM-OO}) = \begin{cases} O\left(\left((n_k^s + a) + n_{k+1}^{-f} - j'\right)^2\right) & n_{k+1}^{-f} \geq (n_k^s + a) \\ O\left((n_k^s + a) \cdot n_{k+1}^{-f}\right) = O\left(n_k^s \cdot n_{k+1}^{-f} + a \cdot n_{k+1}^{-f}\right) & n_{k+1}^{-f} < (n_k^s + a) \end{cases}, \quad (4.63)$$

where $j' \in [1, (n_k^s + a)]$, opposed to $j \in [1, n_k^s]$. From comparing Eqs. (4.62 - 4.63), for the case where the size of added factors is larger than the state, we can deduce that other than the difference between j and j' , they are the same. Judging the second case, we can see they differ by the difference between the size of the state at time k and the number of $\mathcal{A}_{k+1|k+1}$ rows. As we will see later on, OTM-OO empirically proves to be more efficient than OTM, which means that the state at time k is in fact larger than the number of size of $\mathcal{A}_{k+1|k+1}$ rows.

Revisiting Eq. (4.61) in-light of the understanding that the state at time k is in fact larger than the number of $\mathcal{A}_{k+1|k+1}$ rows we can say that OTM is computationally superior to DU without any restricting conditions.

4.5 RESULTS - CONSISTENT DA

In this section we present an extensive analysis of the proposed paradigm for RUB inference under the simplifying assumption of consistent DA and benchmark it against the standard Bayesian inference approach using iSAM₂ efficient methodologies as a proving-ground.

We consider the problem of autonomous navigation and mapping in an unknown environment as a testbed for the proposed paradigm in a simulated environment. The robot performs inference to maintain a belief over its current and past poses and the observed landmarks thus far (i.e. full-SLAM), and uses this belief to decide its next actions within the framework of belief space planning. As mentioned earlier, our proposed paradigm is indifferent to a specific method of inference or decision making.

In order to test the computational effort, we compared inference update using iSAM₂ efficient methodology, once based on the standard Bayesian inference paradigm [36] (here on denoted as iSAM), and second based on our proposed RUB inference paradigm.

All of our complementary methods (see Section 4.4), required to enable inference update based on the RUB inference paradigm, were implemented in MATLAB and are *encased within the inference block*. The iSAM approach uses the GTSAM C++ implementation with the supplied MATLAB wrapper [13]. Considering the general rule of thumb, that MATLAB implementation is at least one order of magnitude slower, the comparison to iSAM as a reference is conservative. All runs were executed on the same Linux machine, with Xeon E3-1241v3 3.5 GHz processor with 32 GB of memory.

In order to get better understanding of the difference between our proposed paradigm and the standard Bayesian inference, we refer to the high-level algorithm diagram given in Figure 4.1.1, which depicts a plan-act-infer framework. Figure 4.1.1a represents a standard Bayesian inference, where only the first inference update iteration is timed for comparison reasons. Figure 4.1.1b shows our novel paradigm RUB inference, while the DA update, along with the first inference update iteration, are being timed for comparison. As the results of this section are under the simplifying assumption of consistent DA, no DA update is required. The computation time comparison is made only over the inference stage, since the rest of the plan-act-infer framework is *identical* in both cases.

As mentioned, our proposed paradigm does not affect estimation accuracy. We verify that in the following experiments, by comparing the estimation results obtained using our approach and iSAM. Both provide essentially the same results in all cases as they are both algebraically identical - share the same factor graph.

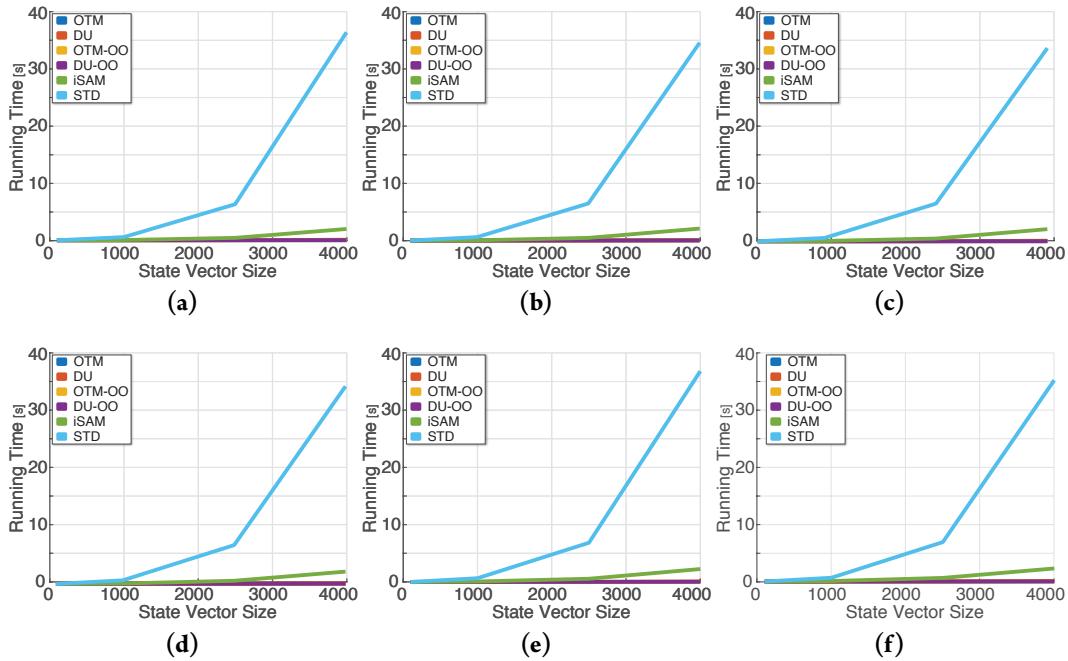


Figure 4.5.1: Method comparison through basic analysis simulation, checking sensitivity to new added measurements and the size of the inference state vector over all the tested methods i.e. STD, iSAM and our four methods, i.e. OTM, UD, OTM-OO and UD-OO. Each graph represents a different number for new rows added to the Jacobian matrix (a) 2 rows (b) 100 rows (c) 200 rows (d) 300 rows (e) 400 rows (f) 500 rows. Due to orders of magnitude issues we also provide zoom-in to our four methods in Figure 4.5.2

4.5.1 BASIC ANALYSIS - SANITY CHECK

The purpose of this experiment is to provide with a basic comparison between the suggested paradigm for RUB inference and the existing standard Bayesian inference. This simulation performs a single horizon BSP calculation, followed by an inference step with a single inference update. The simulation provides a basic analysis of running time for each method, denoted by the *vertical axis*, for a *fully dense information matrix* and with no loop closures. The presented running time is a result of an average between 10^3 repetitions per step per method. Although a fully dense matrix does not represent a real-world scenario, it provides a sufficient initial comparison. The simulation analyzes the sensitivity of each method to the initial state vector size, denoted by the *horizontal axis*, and to the number of new factors, denoted by the different graphs. Since we perform a single horizon step with a single inference update, no re-linearization is necessary; hence, iSAM comparison is valid. The purpose of this check is to provide a simple sensitivity analysis of our methods to state dimension and number of new factors per step, while compared against standard batch update (denoted as STD) and iSAM paradigm. While both STD and iSAM are based on the standard Bayesian inference paradigm, the rest of the methods are based on the novel RUB inference paradigm.

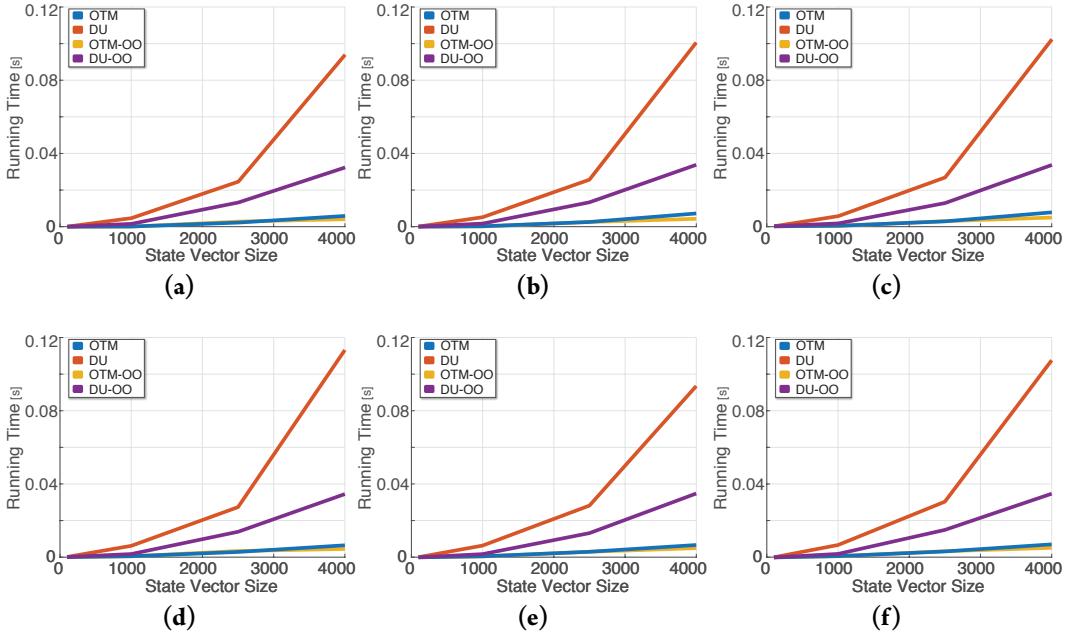


Figure 4.5.2: Zoom-in on Figure 4.5.1, checking sensitivity to new added measurements and the size of the inference state vector over our four methods i.e. OTM, UD, OTM-OO and UD-OO. Each graph represents a different number for new rows added to the Jacobian matrix (a) 2 rows (b) 100 rows (c) 200 rows (d) 300 rows (e) 400 rows (f) 500 rows.

Figure 4.5.1 presents average timing results for all methods, while Figures 4.5.1a - 4.5.1f represent different number of new rows added to the Jacobian matrix (equivalent to adding new measurements), [2 100 200 300 400 500] respectively. After inspecting the results, we found that for all methods, running time is a non-linear, positive-gradient function of the inference state vector size and a linear function of the number of new measurements. Moreover, the running time dependency over the number of new measurements diminish as the inference state vector size grows. For all inspected parameters our methods score the lowest running time with a difference of up to *three orders of magnitude* comparing to iSAM.

Figure 4.5.2 provides a zoom-in of Figure 4.5.1, focusing on our suggested methods. Interestingly while we can clearly see that the OTM methodology is more efficient than the DU method, and the DU-OO is more efficient than DU, no such think can be said on OTM and OTM-OO. From inspecting Figures 4.5.2a - 4.5.2f we can see that up to a state vector size of about 2500 there is no visible difference between OTM and OTM-OO performance, while for larger sizes the latter slightly outperforms the former.

Thus scoring all methods from the fastest to the slowest with a time difference of *four orders of magnitude* between the opposites:

$$\text{OTM-OO} \Rightarrow \text{OTM} \Rightarrow \text{DU-OO} \Rightarrow \text{DU} \Rightarrow \text{iSAM} \Rightarrow \text{STD}$$

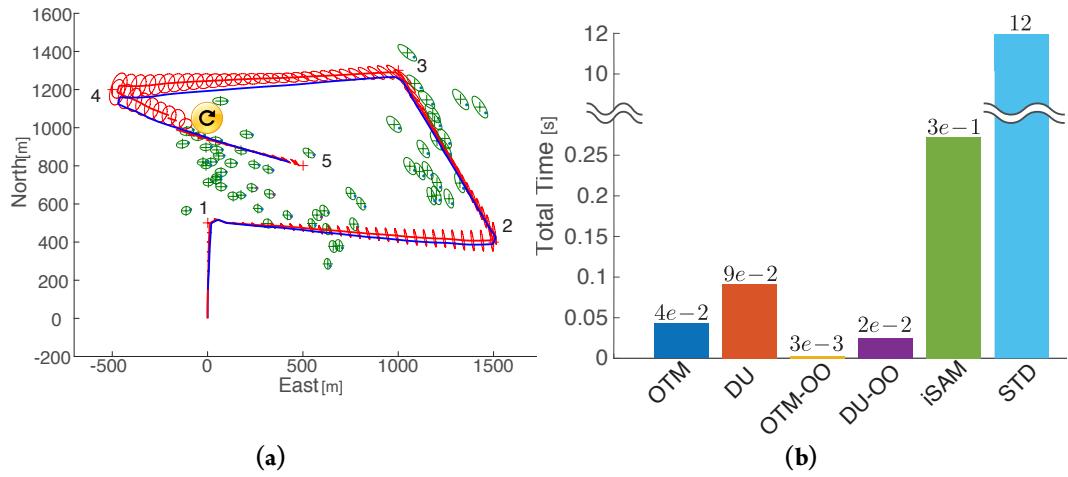


Figure 4.5.3: Second simulation layout and results: (a) The Synthetic Environment, where landmarks are marked in green, targets are numbered and marked with red crosses, the ground truth is denoted by a blue line, the estimated trajectory is denoted by a red line while the covariance is visualized by red ellipse (b) Total average running time of inference update for each method.

4.5.2 BSP IN UNKNOWN ENVIRONMENT - CONSISTENT DA

The purpose of this experiment is to further examine the suggested paradigm of RUB inference, in a real world scenario, under the simplifying assumption of consistent DA. The second simulation performs BSP over continuous action space, in an unknown synthetic environment. In contrast to Section 4.5.1, since now the synthetic environment replicates a real world scenario, the obtained information matrix is now sparse (e.g. Fig. A.1). A robot was given five targets (see Figure 4.5.3a) while all landmarks were a-priori unknown, and was required to visit all targets whilst not crossing a covariance value threshold. The largest loop closure in the trajectory of the robot, and the first in a series of large loop closures, is denoted by a yellow \circlearrowleft sign across all relevant graphs. The robot performs BSP over continuous action space, with a finite horizon of five look ahead steps [29]. During the inference update stage each of the aforementioned methods were timed performing the first inference update step. It is worth mentioning that our paradigm is agnostic to the specific planning method or whether the action space is discrete or continuous.

The presented running time is a result of an average between 10^3 repetitions per step per method. Similarly to Section 4.5.1, as can be seen in Figure 4.5.3b, the suggested MATLAB implemented methods are up to *two orders of magnitude* faster than iSAM used in a MATLAB C++ wrapper. Interestingly, the use of sparse information matrices changed the methods' timing hierarchy. While OTM-OO still has the best timing results (3×10^{-3} sec), *two orders of magnitude faster than iSAM*, OTM and DU-OO switched places. So the timing hierarchy from fastest to slowest is:

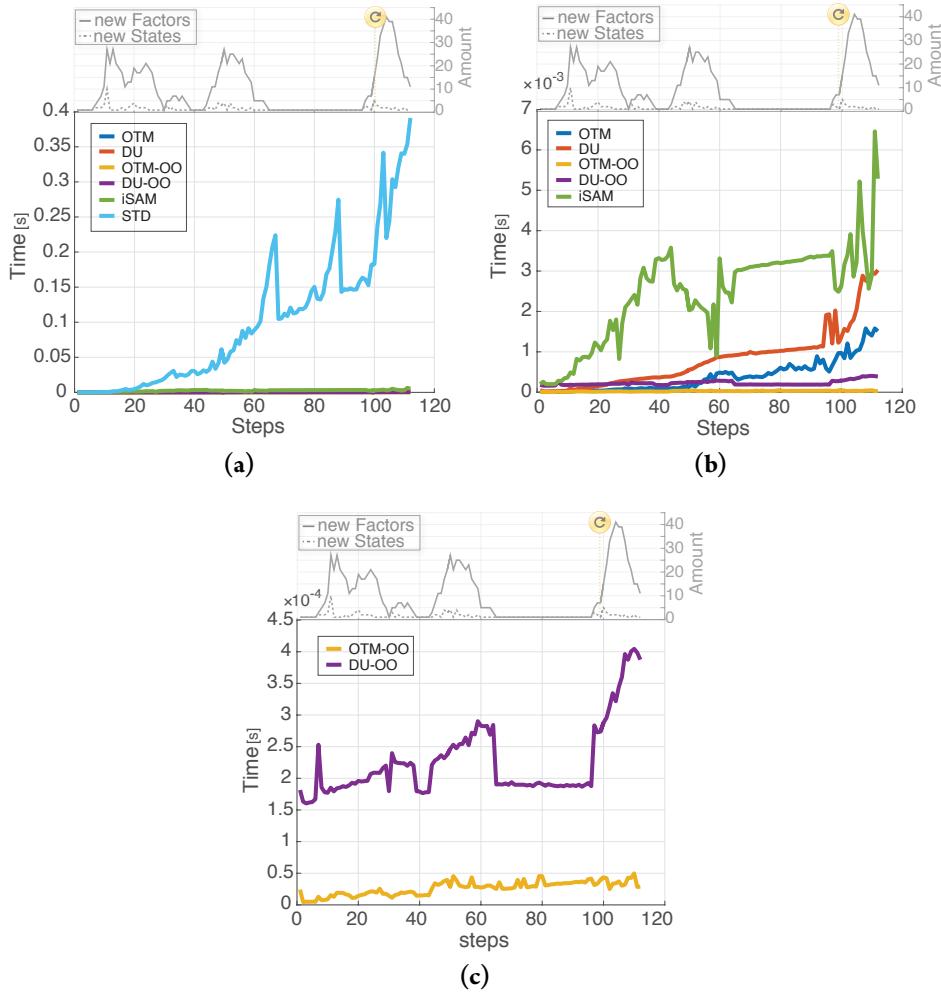


Figure 4.5.4: Second simulation timing results for the scenario presented in Figure 4.5.3a. Upper part of each graph provides indication on new factors and new states per computation step while the lower presents the methods timing results: (a) All six methods (b) OTM, DU, OTM-OO, DU-OO and iSAM methods (c) OTM-OO and DU-OO methods.

$$\text{OTM-OO} \Rightarrow \text{DU-OO} \Rightarrow \text{OTM} \Rightarrow \text{DU} \Rightarrow \text{iSAM} \Rightarrow \text{STD}$$

After demonstrating the use of our novel paradigm drastically reduce cumulative running time, we continue on to showing that in a few aspects it is also less sensitive. Figure 4.5.4 presents the performance results of each of the methods per simulation step. The upper graphs presents the number of new factors and new states per each step, while the lower graph presents the average running time of each method as a function of the simulation step. The \circlearrowleft sign, represents the first largest loop closure in a series of large loop closures. While some of the behavior presented in Figure 4.5.4 can be related to machine noise, from carefully inspecting Figure 4.5.4, alongside the trajectory of the robot in Figure 4.5.3a, a few interesting observations can still be made. The first observation relates to the "flat

line” area noticeable in the upper graph of Figure 4.5.4b between time steps 60 – 90. This time steps range is equivalent to the path between the third and fourth targets, were the only factor added to the belief is motion based. As a result, a single new state (the new pose) is presented to the belief, along with a single motion factor. In this range, the timing results of iSAM DU and OTM present a linear behavior with a relatively small gradient. This gradient is attributed to the computational effort of introducing a single factor, containing a new state, to the belief. While the vertical difference between the aforementioned can be attributed to the sensitivity of each method to the number of states and factors in the belief.

From this observation, we can try to better understand the reason for the substantial time difference between the methods. Basing a method on RUB inference, rather than on standard Bayesian inference, will not magically change the computational impact of introducing factors or new states to the belief. However, because RUB inference is re-using calculations from precursory planning, the computational burden is being “paid” once, rather than twice as in the standard Bayesian inference. For the simple example of strictly motion propagation, since this motion based factor has already been introduced during precursory planning, under RUB inference it offers no additional computational burden. In the same manner, the reason RUB inference is less sensitive to the state dimensionality originates in calculations re-use. Under incremental update performed by iSAM, the state dimension is mostly noticeable when in need of re-ordering and/or re-eliminating states. Although same mechanisms also affect RUB inference, our method avoids them whenever they were adequately performed during the precursory planning, thus reducing inference computation time.

Another interesting observation refers to “pure” loop closures, were there are measurements with no addition of new variables to the state vector, i.e. measurements to previously observed landmarks. For the case of “pure” loop closures, STD, iSAM and the DU based methods (i.e. DU and DU-OO) experienced the largest timing spikes throughout the trajectory of each method while both OTM based methods experienced minor spikes if any.

By introducing the OO methodology to both DU and OTM, we drastically reduce the methods sensitivity to the motion propagation e.g. the once-positive gradient line in DU during time steps 60 – 90, turned into a flat line in DU-OO as can easily be seen in Figure 4.5.4c. Moreover, while both DU and OTM present some sensitivity to different occurrences, i.e. the size of the state vector, new measurements and loop closures, this sensitivity is drastically reduced by introducing the OO methodology, e.g. OTM-OO is basically a flat line throughout the simulation as can easily be seen in Figure 4.5.4.

In conclusion, our methods, based on RUB inference, particularly OTM-OO, seem to be more resilient to large loop closures that were already detected during planning, state vector size, belief size, number of newly added measurements or even the combination of the aforementioned.

4.6 RUBI: RE-USE BSP FOR INFERENCE UPDATE

In order to address the more general and realistic scenario, the DA might require correction before proceeding to update the new acquired measurements. In the sequel we cover the possible scenarios of inconsistent data association and its graphical materialization, followed by a paradigm to update inconsistent DA from planning stage according to the actual DA attained in the consecutive inference stage. We later examine both the computational aspects and the sensitivity of the paradigm to various parameters both on simulated and real-life data.

4.6.1 TYPES OF INCONSISTENT DA

We would now discuss, without losing generality, the actual difference between the two aforementioned beliefs $b[X_{k+1|k}]$ and $b[X_{k+1|k+1}]$. As already presented in Section 4.4, in case of a consistent DA i.e. $\mathcal{M}_{k+1|k} = \mathcal{M}_{k+1|k+1}$, the difference between the two beliefs is narrowed down to the RHS vectors $d_{k+1|k}$ and $d_{k+1|k+1}$ which encapsulates the measurements $z_{k+1|k}$ and $z_{k+1|k+1}$ respectively. However, in the real world it is possible that the DA predicted in precursory planning would prove to be inconsistent to the DA attained in inference.

There are six possible scenarios representing the relations between DA in inference and precursory planning:

- In planning, association is assumed to either a new or existing variable, while in inference no measurement is received.
- In planning it is assumed there will be no measurement to associate to, while in inference a measurement is received and associated to either a new or existing variable.
- In planning, association is assumed to an existing variable, while in inference it is to a new variable.
- In planning, association is assumed to a new variable, while in inference it is to an existing variable.
- In planning, association is assumed to an existing variable, while in inference it is also to an existing variable (whether the same or not).
- In planning, association is assumed to a new variable, while in inference it is also to a new variable (whether the same or not).

While the first four bullets always describe inconsistent DA situations (e.g. in planning we assumed a known tree would be visible but instead we saw a new bench, or vice versa), the last two bullets may provide consistent DA situations. In case associations in planning and in inference are to the same (un)known variables we would have a consistent DA.

While different planning paradigms might diminish occurrences of inconsistent DA, e.g. by better predicting future associations, none can avoid it completely. Methods to better predict future observations/associations will be investigated in future work, potentially leveraging Reinforcement Learning (RL) techniques. As mentioned in Section 4.3.2, in this paper we do not predict occurrences of new landmarks, hence every new landmark in inference would result in inconsistent DA.

In the following section we provide a method to update inconsistent DA, regardless of a specific inconsistency scenario or a solution paradigm. This method utilizes the incremental methodologies of iSAM₂ [36] in order to efficiently update the belief from the planning stage to have consistent DA with the succeeding inference.

4.6.2 UPDATING INCONSISTENT DA

Inconsistent DA can be interpreted as disparate connections between variables. As discussed earlier, these connections, denoted as factors, manifest in rows of the Jacobian matrix or in factor nodes of a FG. Two FGs with different DA would thus have different graph topology. We demonstrate the inconsistent DA impact over graph topology using the example presented in Figure 4.6.1: Figure 4.6.1a represents the belief $b[X_{k+1|k}]$ from planning stage, and Figure 4.6.1b represents the belief $b[X_{k+1|k+1}]$ from the inference stage. Even-though the same elimination order is used, the inconsistent DA would also create a different topology between the resulting BTs, e.g. the resulting BTs for the aforementioned FGs are Figure 4.6.1d and Figure 4.6.1e accordingly.

Performing action $u_{k|k+1}$, provides us with new measurements $z_{k+1|k+1}$, which are gathered to the factor set $\{f_j\}_{k+1|k+1}$ (see Appendix-B for factor definition). From the precursory planning stage we have the belief $b[X_{k+1|k}]$ along with the corresponding factor set $\{f_i\}_{k+1|k}$ for time $k + 1$. Since we performed inference over this belief during the planning stage, we have already eliminated the FG, denoted as $\mathcal{FG}_{k+1|k}$, into a BT denoted as $\mathcal{T}_{k+1|k}$, e.g. see Figure 4.6.1a and Figure 4.6.1d, respectively.

We would like to update both the FG $\mathcal{FG}_{k+1|k}$ and the BT $\mathcal{T}_{k+1|k}$ from the planning stage, using the new factors $\{f_j\}_{k+1|k+1}$ from the inference stage. Without losing generality we use Figure 4.6.1 to demonstrate and explain the DA update process. Let us consider all factors of time $k + 1$ from both planning $\{f_i\}_{k+1|k}$ and inference $\{f_j\}_{k+1|k+1}$. We can divide these factors into three categories:

The first category contains factors with consistent DA - Good Factors. These factors originate from only the last two DA scenarios, in which both planning and inference considered either the same ex-

Variable	Description
$\square_{t k}$	Of time t while current time is k
$\mathcal{FG}_{t k}$	Factor graph (FG) at time t
$\mathcal{T}_{t k}$	Bayes Tree (BT) at time t
$\mathcal{M}_{t k}$	Data Association (DA) at time t
\mathcal{M}_t^{\cap}	Consistent DA at time t
\mathcal{M}_t^{rmv}	DA at time t from planning inconsistent with inference, indicating factors to be removed
\mathcal{M}_t^{add}	DA at time t from inference inconsistent with planning, indicating factors to be added
$\{f_r\}_t^{rmv}$	Factors at time t from planning inconsistent with inference, to be removed
$\{f_s\}_t^{add}$	Factors at time t from inference inconsistent with planning, to be added
$\{X\}_t^{inv}$	All states at time t , involved in $\{f_r\}_t^{rmv}$ and $\{f_s\}_t^{add}$
\mathcal{T}_t^{inv}	Sub-BT of $\mathcal{T}_{t k}$ composed of all cliques containing $\{X\}_t^{inv}$
$\{X\}_t^{inv*}$	All states at time t , related to the sub-BT \mathcal{T}_t^{inv}
\mathcal{FG}_t^{inv}	The detached part of $\mathcal{FG}_{t k}$ containing $\{X\}_t^{inv*}$
\mathcal{FG}_t^{upd}	The FG \mathcal{FG}_t^{inv} after DA update
\mathcal{T}_t^{upd}	The sub-BT eliminated from \mathcal{FG}_t^{upd}
$\mathcal{FG}_{t k}^{upd}$	The Factor Graph at time t with all-correct DA
$\mathcal{T}_{t k}^{upd}$	The Bayes Tree at time t with all-correct DA

Table 4.1: Notations for Section 4.6.2

isting variable or a new one. Consistent DA factors do not require our attention (other than updating the measurements in the RHS vector). Indices of consistent DA factors can be obtained by intersecting the DA from planning with that of inference:

$$\mathcal{M}_{k+1}^{\cap} = \mathcal{M}_{k+1|k} \cap \mathcal{M}_{k+1|k+1}. \quad (4.64)$$

The second category - Wrong Factors, contains factors from planning stage with inconsistent DA to inference, which therefore should be removed from $\mathcal{FG}_{k+1|k}$. These factors can originate from all DA scenarios excluding the second. Indices of inconsistent DA factors from planning, can be obtained by calculating the relative complement of $\mathcal{M}_{k+1|k}$ with respect to $\mathcal{M}_{k+1|k+1}$:

$$\mathcal{M}_{k+1}^{rmv} = \mathcal{M}_{k+1|k} \setminus \mathcal{M}_{k+1|k+1}. \quad (4.65)$$

The third category - New Factors, contains factors from the inference stage with inconsistent DA to planning; hence, these factors should be added to $\mathcal{FG}_{k+1|k}$. These factors can originate from all DA scenarios excluding the first. Indices of inconsistent DA factors from inference, can be obtained by calculating the relative complement of $\mathcal{M}_{k+1|k+1}$ with respect to $\mathcal{M}_{k+1|k}$:

$$\mathcal{M}_{k+1}^{add} = \mathcal{M}_{k+1|k+1} \setminus \mathcal{M}_{k+1|k}. \quad (4.66)$$

We now use our example from Figure 4.6.1 to illustrate these different categories:

- The first category - Good Factors, contains all factors from time $k+1$ that appear both in Figure 4.6.1a and 4.6.1b, i.e. the motion model factor between x_k to x_{k+1} .
- The second category - Wrong Factors, contains all factors that appear only in Figure 4.6.1a, i.e. the star marked factor in Figure 4.6.1a. In this case the inconsistent DA is to an existing variable, landmark l_j , was considered to be observed in planning but is not seen in the succeeding inference.
- The third category - New Factors, contains all factors that appear only in Figure 4.6.1b, i.e. the star marked factors in Figure 4.6.1b. In this case the inconsistent DA is both to an existing and a new variable. Instead of landmark l_j that was considered to be observed in planning, a different existing landmark l_i has been seen, along with a new landmark l_r .

Once the three aforementioned categories are determined, we use iSAM2 methodologies, presented in [36], to incrementally update $\mathcal{FG}_{k+1|k}$ and $\mathcal{T}_{k+1|k}$, see Alg. 1. The involved factors are denoted by all factors from planning needed to be removed (Wrong Factors), and all factors from inference needed to be added (New Factors),

$$\{f_r\}_{k+1}^{rmv} = \prod_{r \in \mathcal{M}_{k+1}^{rmv}} f_r \quad , \quad \{f_s\}_{k+1}^{add} = \prod_{s \in \mathcal{M}_{k+1}^{add}} f_s. \quad (4.67)$$

The involved variables, denoted by $\{X\}_{k+1}^{inv}$, are all variables related to the factor set $\{f_r\}_{k+1}^{rmv}$ and the factor set $\{f_s\}_{k+1}^{add}$ (Alg. 1, line 6), e.g. the colored variables in Figures 4.6.1a and 4.6.1b accordingly. In $\mathcal{T}_{k+1|k}$, all cliques between the ones containing $\{X\}_{k+1}^{inv}$ up to the root are marked and denoted as the involved cliques, e.g. colored cliques in Figure 4.6.1d. The involved cliques are detached and denoted by $\mathcal{T}_{k+1}^{inv} \subset \mathcal{T}_{k+1|k}$ (line 7). This sub-BT \mathcal{T}_{k+1}^{inv} , contains more variables than just $\{X\}_{k+1}^{inv}$. The involved variable set $\{X\}_{k+1}^{inv}$, is then updated to contain all variables from \mathcal{T}_{k+1}^{inv} and denoted by $\{X\}_{k+1}^{inv*}$ (line 8).

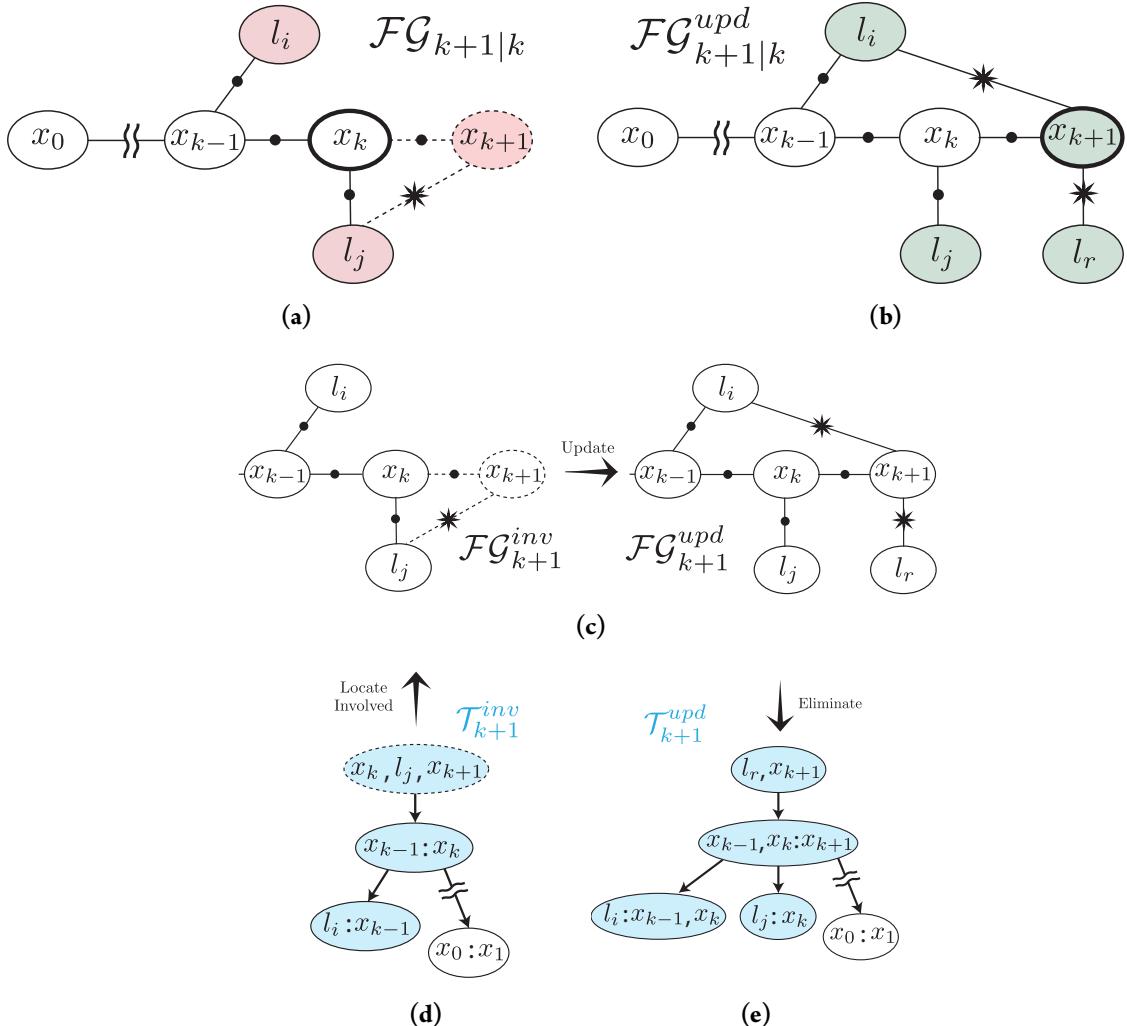


Figure 4.6.1: The process of incremental DA update, following on iSAM2 methodologies. (a) and (b) show factor graphs for $b[X_{k+1|k}]$ and $b[X_{k+1|k+1}]$, respectively, which differ due to incorrect association considered in the planning phase - l_i was predicted to be observed within planning, while in practice l_i and l_r were observed at time instant $k + 1$. In (a), current-time robot pose is bolded, horizon factors and states are dotted. Involved variables from DA comparison are marked in red in (a) and green in (b). The belief $b[X_{k+1|k}]$, represented by a Bayes tree shown in (d), is divided in two: sub Bayes tree containing all involved variables and parent cliques up to the root (marked in blue) and the rest of the Bayes tree in white. The former sub Bayes tree is re-eliminated by (i) forming the corresponding portion of the factor graph, as shown in the left figure of (c); (ii) removing incorrect DA and adding correct DA factors, which yields the factor graph shown in the right figure of (c); (iii) re-eliminating that factor graph into a sub Bayes tree, marked blue in (e), and re-attaching the rest of the Bayes tree. While the obtained Bayes tree now has a correct DA, it is conditioned on (potentially) incorrect measurement values for consistent-DA factors, which therefore need to be updated to recover the posterior belief $b[X_{k+1|k+1}]$.

The part of $\mathcal{FG}_{k+1|k}$, that contains all involved variables $\{X\}_{k+1}^{inv*}$ is detached and denoted by \mathcal{FG}_{k+1}^{inv} (line 9). While \mathcal{T}_{k+1}^{inv} is the corresponding sub-BT to the acquired sub-FG \mathcal{FG}_{k+1}^{inv} .

In order to finish updating the DA, all that remains is updating the sub-FG \mathcal{FG}_{k+1}^{inv} with the correct DA and re-eliminate it to get an updated BT. All factors $\{f_r\}_{k+1}^{rmv}$ are removed from \mathcal{FG}_{k+1}^{inv} , then all factors $\{f_r\}_{k+1}^{add}$ are added (line 10). The updated sub-FG is denoted by \mathcal{FG}_{k+1}^{upd} , e.g. update illustration

Algorithm 1 - Data Association Update

```

1: function UPDATEDA( $\mathcal{FG}_{k+1|k}$ ,  $\mathcal{M}_{k+1|k}$ ,  $\mathcal{FG}_{k+1|k+1}$ ,  $\mathcal{M}_{k+1|k+1}$ )
2:    $\mathcal{M}_{k+1}^{rmv} \leftarrow \mathcal{M}_{k+1|k} \setminus \mathcal{M}_{k+1|k+1}$                                  $\triangleright$  indices of factors required to be removed
3:    $\mathcal{M}_{k+1}^{add} \leftarrow \mathcal{M}_{k+1|k+1} \setminus \mathcal{M}_{k+1|k}$                              $\triangleright$  indices of factors required to be added
4:    $\{f_r\}_{k+1}^{rmv} \leftarrow \prod_{r \in \mathcal{M}_{k+1}^{rmv}} \{f_r\}_{k+1}$                        $\triangleright$  factors required to be removed
5:    $\{f_s\}_{k+1}^{add} \leftarrow \prod_{s \in \mathcal{M}_{k+1}^{add}} \{f_s\}_{k+1}$                        $\triangleright$  factors required to be added
6:    $\{X\}_{k+1}^{inv} \leftarrow \text{Variables}(\{f_r\}_{k+1}^{rmv}) \cup \text{Variables}(\{f_s\}_{k+1}^{add})$      $\triangleright$  get involved variables
7:    $\mathcal{T}_{k+1}^{inv} \leftarrow \mathcal{T}_{k+1|k}^{\{X\}_{k+1}^{inv}}$                                           $\triangleright$  get corresponding sub-BT
8:    $\{X\}_{k+1}^{inv*} \xleftarrow{\text{get all variables}} \mathcal{T}_{k+1}^{inv}$                             $\triangleright$  update involved variables
9:    $\mathcal{FG}_{k+1}^{inv} \leftarrow \mathcal{FG}_{k+1|k}^{\{X\}_{k+1}^{inv*}}$                                 $\triangleright$  get corresponding sub-FG
10:   $\mathcal{FG}_{k+1}^{upd} \leftarrow [\mathcal{FG}_{k+1}^{inv} \setminus \{f_r\}_{k+1}^{rmv}] \cup \{f_s\}_{k+1}^{add}$        $\triangleright$  Update the sub Factor Graph
11:   $\mathcal{T}_{k+1}^{upd} \xleftarrow{\text{eliminate}} \mathcal{FG}_{k+1}^{upd}$                                       $\triangleright$  re-eliminate the updated sub-FG into BT
12:   $\mathcal{FG}_{k+1|k}^{upd} \leftarrow [\mathcal{FG}_{k+1|k} \setminus \mathcal{FG}_{k+1}^{inv}] \cup \mathcal{FG}_{k+1}^{upd}$        $\triangleright$  Update the Factor Graph
13:   $\mathcal{T}_{k+1|k}^{upd} \leftarrow [\mathcal{T}_{k+1|k} \setminus \mathcal{T}_{k+1}^{inv}] \cup \mathcal{T}_{k+1}^{upd}$             $\triangleright$  Update the Bayes Tree
14:  return  $\mathcal{FG}_{k+1|k}^{upd}$ ,  $\mathcal{T}_{k+1|k}^{upd}$ .
15: end function

```

in Figure 4.6.1c.

By re-eliminating \mathcal{FG}_{k+1}^{upd} , a new updated BT, denoted by \mathcal{T}_{k+1}^{upd} , is obtained (line 11), e.g. the colored sub-BT in Figure 4.6.1e. This BT is then re-attached back to $\mathcal{T}_{k+1|k}$ instead of \mathcal{T}_{k+1}^{inv} , subsequently the new BT is now with consistent DA and is denoted as $\mathcal{T}_{k+1|k}^{upd}$ (line 13). In a similar manner $\mathcal{FG}_{k+1|k}^{upd}$ is obtained by re-attaching \mathcal{FG}_{k+1}^{upd} instead of \mathcal{FG}_{k+1}^{inv} to $\mathcal{FG}_{k+1|k}$ (line 12). At this point the DA in both the FG and the BT is fixed. For example, by completing the aforementioned steps, Figures 4.6.1a and 4.6.1d will have the same topology as Figures 4.6.1b and 4.6.1e.

After the DA update, the BT $\mathcal{T}_{k+1|k}^{upd}$ has consistent DA to that of $\mathcal{M}_{k+1|k+1}$. However, it is still not identical to $\mathcal{T}_{k+1|k+1}$ due to difference between measurement values predicted in planning to the values obtained in inference. The DA update dealt with inconsistent DA factors and their counterparts. For these factors the new measurements from inference were updated in the corresponding RHS vector values within the BT. The consistent DA factors, on the other hand, were left untouched; therefore, these factors do not contain the new measurement values from inference but measure-

ment values from the planning stage instead. These inconsistent measurements are thus baked into the RHS vector $d_{k+1|k}$ and in the appropriate cliques of the BT $\mathcal{T}_{k+1|k}^{upd}$. In order to update the RHS vector $d_{k+1|k}$, or equivalently update the corresponding values within relevant cliques of the BT, one can use any of the methods presented in Section 4.4.

4.7 RESULTS - RUBI

In this section we present an extensive analysis of the proposed paradigm for RUB Inference and benchmark it against the standard Bayesian inference approach using iSAM2 efficient methodologies as a proving-ground.

We consider the problem of autonomous navigation and mapping in an unknown environment as a testbed for the proposed paradigm, first in a simulated environment and later-on in a real-world environment (as discussed in the sequel). The robot performs inference to maintain a belief over its current and past poses and the observed landmarks thus far (i.e. full-SLAM), and uses this belief to decide its next actions within the framework of belief space planning. As mentioned earlier, our proposed paradigm is indifferent to a specific method of inference or decision making.

In order to test the computational effort, we compared inference update using iSAM2 efficient methodology, once based on the standard Bayesian inference paradigm [36] (here on denoted as iSAM), and second based on our proposed RUB Inference paradigm.

All scripts required to enable inference update based on the RUB Inference paradigm, were implemented in MATLAB and are *encased within the inference block*. The iSAM approach uses the GT-SAM C++ implementation with the supplied MATLAB wrapper [13]. Considering the general rule of thumb, that MATLAB implementation is at least one order of magnitude slower, the comparison to iSAM as a reference is conservative. All runs were executed on the same Linux machine, with Xeon E3-1241v3 3.5 GHz processor with 32 GB of memory.

In order to get better understanding of the difference between our proposed paradigm and the standard Bayesian inference, we refer to the high-level algorithm diagram given in Figure 4.1.1, which depicts a plan-act-infer framework. Figure 4.1.1a represents a standard Bayesian inference, where only the first inference update iteration is timed for comparison reasons. Figure 4.1.1b shows our novel paradigm RUB Inference, while the DA update, along with the first inference update iteration, are being timed for comparison. The computation time comparison is made only over the inference stage, since the rest of the plan-act-infer framework is *identical* in both cases.

As mentioned, our proposed paradigm does not affect estimation accuracy. We verify that in the following experiments, by comparing the estimation results obtained using our approach and iSAM. Both provide essentially the same results in all cases; we provide an explicit accuracy comparison

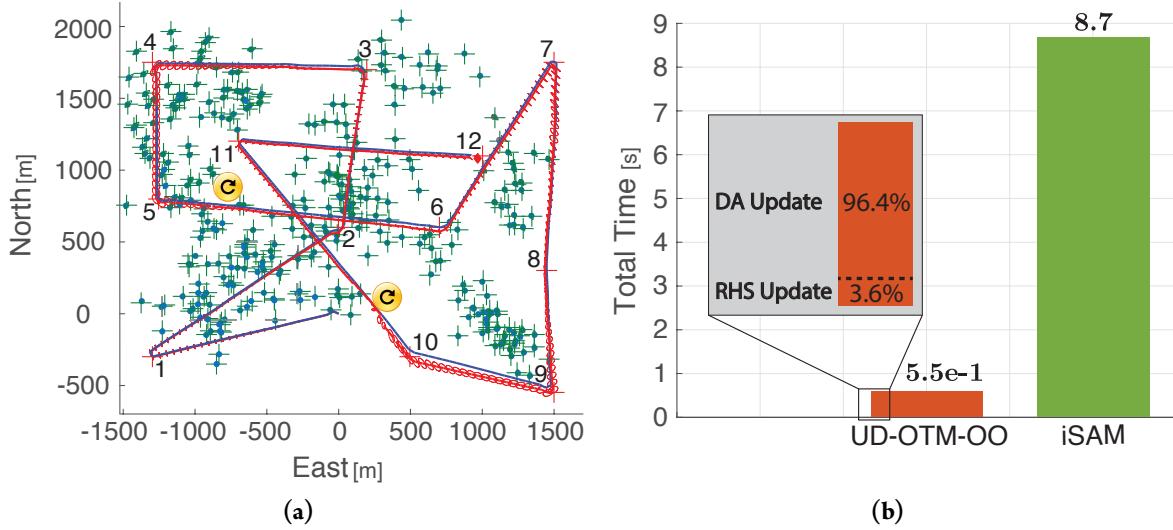


Figure 4.7.1: Simulation layout and results: (a) The Synthetic Environment, where landmarks are marked in green, targets are numbered and marked with red crosses, the ground truth is denoted by a blue line, the estimated trajectory is denoted by a red line while the covariance is visualized by red ellipse. (b) Total average running time of inference update for each method, when 50% of the steps were with inconsistent DA.

with real-world data experiment (Section 4.7.2).

4.7.1 RELAXING THE CONSISTENT DA ASSUMPTION

The purpose of this experiment is to further examine the suggested paradigm of RUB Inference, in a real world scenario, while relaxing the simplifying assumption of consistent DA. The third simulation performs BSP over continuous action space, in an unknown synthetic environment. A robot was given twelve targets (see Figure 4.7.1a) while all landmarks were a-priori unknown, and was required to visit all targets whilst not crossing a covariance value threshold. The experiments presented in Section 4.5 were based on the simplifying assumption of consistent DA between inference and precursory planning, which can often be violated in real world scenarios. In this simulation we relax this restricting assumption and test our novel paradigm under the more general case where DA might be inconsistent.

The main reason for inconsistent data association lies in the perturbations caused by imperfect system and environment models. These perturbations increase the likelihood of inconsistent DA between inference and precursory planning. While the planning paradigm uses state estimation to decide on future associations, the further it is from the ground truth the more likely for inconsistent DA to be received. This imperfection is modeled by formulating uncertainty in all models (see Section 4.2).

For a more conservative comparison, in addition to the aforementioned, we force inconsistent DA between inference and precursory planning for all new variables. In contrast to planning paradigms that can provide DA to new variables, in addition to an unknown map, the robot's planning paradigm considers only previously-mapped landmarks. As a result of this limitation, the DA received from the planning stage can not offer new landmarks to the state vector. Consequently, each new landmark would essentially mean facing inconsistent DA, while the single scenario in which a consistent DA is obtained (see Section 4.6), occurs when both planning and inference are considering the same known landmark. Both perturbations caused by uncertainty and considering only previously mapped landmarks, resulted in just 50% DA consistency between planning and succeeding inference in this experiment.

Following the findings of Section 4.5, out of the four suggested methods we choose to continue the comparison just with the OTM-OO method. While OTM-OO assumed consistent DA, the more general approach deals with inconsistent DA before updating the RHS vector. We denote the complete approach, updating DA followed by OTM-OO, as UD-OTM-OO, where UD stands for Update Data association. It is important to clarify that UD-OTM-OO and for consistent DA also OTM-OO, yield the same estimation accuracy as iSAM, since the inference update using RUB inference results in the same topological graph with the same values. Such comparison will be presented later on using a real-world data in Section 4.7.2. For that reason, the accuracy aspect will not be discussed further in this section. While the scenario presented in Figure 4.7.1a contains at least ten large loop closures, for the readers convenience we marked two of them using yellow \odot signs. Same loop closures are also marked in Figure 4.7.2 for comparison.

Figure 4.7.1b presents the cumulative computation time of the inference update phase throughout the simulation. We can see that the majority of UD-OTM-OO computation time, i.e. 96.4%, is dedicated to DA update while only 3.6% for updating the RHS vector. Although the need for DA update increased running time (as to be expected), UD-OTM-OO still outperforms iSAM by an order of magnitude.

In addition to the improvement in total computation time of the inference update stage, we continue on analyzing the "per step" behavior of UD-OTM-OO, and demonstrate that in a few aspects it is less sensitive than iSAM. Figure 4.7.2a presents per step computation time of both UD-OTM-OO and iSAM, as well as the RHS update running time of UD-OTM-OO. Our suggested paradigm not only outperforms iSAM in the cumulative computation time, but also outperforms it for each individual step. While Figure 4.7.2a presents the difference in average computation time per-step, Figures 4.7.2b and 4.7.2c capture the reason for this difference as suggested in Section 4.4. Figure 4.7.2c presents the number of added factors in iSAM denoted by a green line, as opposed to in UD-OTM-

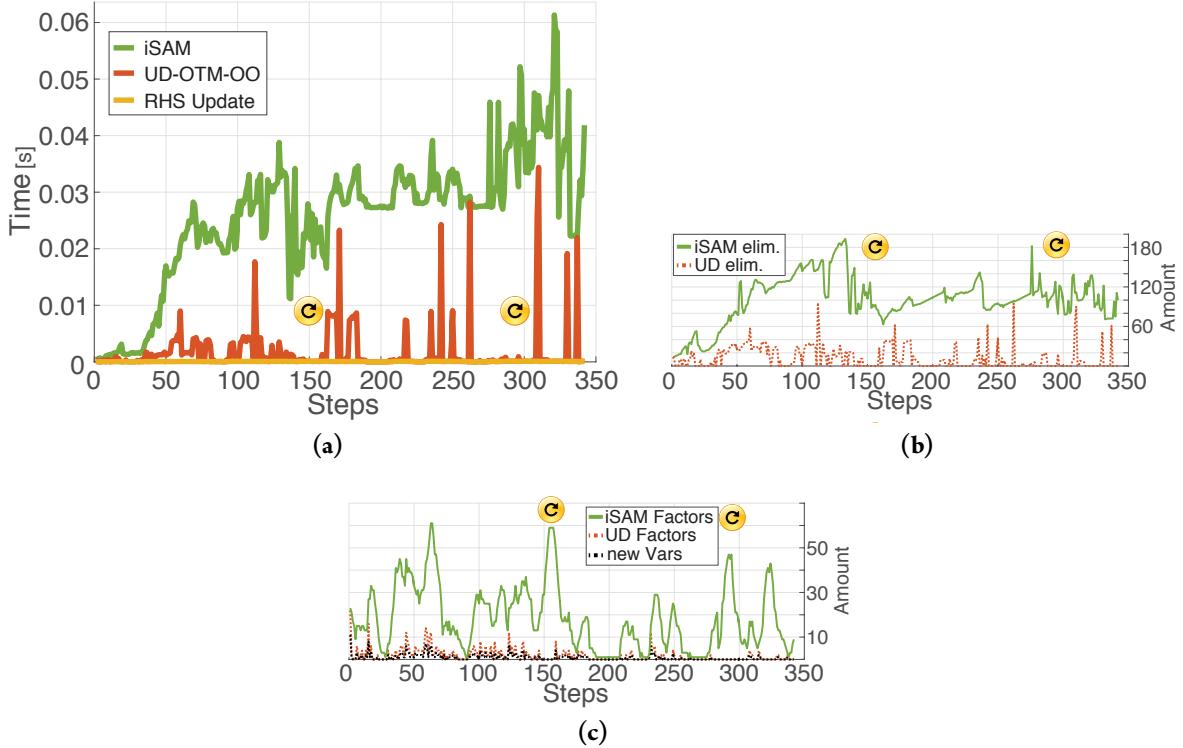


Figure 4.7.2: Per-step analysis of the simulation presented in Figure 4.5.3. In 50% of the steps, planning and succeeding inference are with consistent DA: (a) Per-step timing results of iSAM performing standard Bayesian inference in green, UD-OTM-OO performing RUB inference in orange and the RHS update portion out of UD-OTM-OO in yellow. (b) Number of eliminations per-step, in the inference update stage for both iSAM and UD-OTM-OO. (c) Number of newly added factors in iSAM per step, newly added factors in UD-OTM-OO per step, and the number of new variables introduced to the belief per step.

OO denoted by an orange line, and the number of new variables per step denoted by a black line. Figure 4.7.2b presents the number of eliminations made during inference update in both methods. Number of eliminations reflects the number of involved variables in the process of converting FG into a BT (see Appendix A and Algorithm 1 line 11 for the equivalent processes in iSAM and UD-OTM-OO accordingly).

After carefully inspecting both figures, alongside the robot's trajectory in Figure 4.7.1a, the following observations can be made. Even with the limitation over the planning paradigm, both the number of new factors added and the number of re-eliminations during the inference update stage, are substantially smaller than their iSAM counterparts. These large differences are some of the reasons for UD-OTM-OO's better performance. Due to the limitation over the planning paradigm, new observation factors (i.e. new landmarks added each step) in both iSAM and UD-OTM-OO are identical. While in iSAM new observation factors constitute a small fraction of total factors, for UD-OTM-OO, they constitute more than half of total factors. After comparing the re-elimination graph with

the timing results for each of the methods, it appears both trends and peaks align, so we assume UD-OTM-OO as well as iSAM to be mostly sensitive to the amount of re-eliminations (further analysis is required).

Both re-elimination and added factors amounts, can be further reduced by smart reordering and relaxing the limitation over the planning paradigm accordingly.

As observed in Section 4.4, our method seems to be more resilient to loop closures. By inspecting the yellow \odot signs in Figure 4.7.2c, we can see that in both cases, iSAM introduce around 50 factors of previously known variables (i.e. the black line representing new variables is zeroed), while UD-OTM-OO introduces no factors at all. These two loop closure examples beautifully demonstrate the advantage of using RUB Inference. For cases of consistent or partially consistent DA, when encountering a loop closure (i.e. observing a previously mapped landmark) our method saves valuable computation time since loop closures are only calculated once, in the planning stage (e.g. see timing response for loop closure at the appropriate yellow \odot signs in Figure 4.7.2a).

Our method also seems to be less sensitive to state dimensionality. Inspecting steps 192 – 208 and 263 – 275 in Figure 4.7.2c, we observe there are no new factors, i.e. the computation time is a result of motion factors; inspecting Figure 4.7.2a we observe that in spite of the aforementioned, iSAM computation time is much larger than our method. From this comparison we can infer our suggested method is less sensitive to state dimensionality. As explained earlier, this originates in the reduced number of re-eliminations and state re-ordering in RUB inference when compared to iSAM, e.g. when the amount of re-eliminations in Figure 4.7.2b is almost the same between the two (like in steps 171, 262, 310), the equivalent computation time in Figure 4.7.2a is also almost identical.

4.7.2 REAL-WORLD EXPERIMENT USING KITTI DATASET

After the promising performance in a simulated environment, we tested our paradigm for inference update via BSP in a real-world environment using KITTI dataset [21]. The KITTI dataset, recorded in the city of Karlsruhe, contains stereo images, Laser scans and GPS data. For this work, we used the raw images of the left stereo camera, from the Residential category file: 2011_10_03_drive_0027, as measurements, as well as the supplied ground truth for comparison.

In this experiment we consider a robot, equipped with a single monocular camera, performing Active Full-SLAM in the previously unknown streets of Karlsruhe Germany. The robot starts with a prior over its initial pose and with no prior over the environment. At time k the robot executes BSP on the single step action sequence taken in the KITTI dataset at time $k + 1$. At the end of each BSP session, the robot executes the chosen action, and receives measurements from the KITTI dataset. Inference update is then being performed in two separate approaches, the first following the standard

Bayesian inference approach and the second following our proposed RUB inference approach. The inference update following each is compared for computation time and accuracy. The following sections explain in-detail how planning and perception are being executed in this experiment.

EXPERIMENT PARAMETERS

For the readers convenience, this section covers all the parameters used for this experiment and were not provided by KITTI.

Prior belief standard deviation	$\begin{bmatrix} 1^o \cdot I_{3 \times 3} & \mathbf{0} \\ \mathbf{0} & 1_{[m]} \cdot I_{3 \times 3} \end{bmatrix}$
Motion Model standard deviation	$\begin{bmatrix} 0.5^o \cdot I_{3 \times 3} & \mathbf{0} \\ \mathbf{0} & 0.5_{[m]} \cdot I_{3 \times 3} \end{bmatrix}$
Observation Model standard deviation	$\begin{bmatrix} 1_{[px]} & \mathbf{0} \\ \mathbf{0} & 1_{[px]} \end{bmatrix}$
Camera Aperture	90^o
Camera acceptable Sensing Range	between $2_{[m]}$ and $40_{[m]}$

Table 4.1: Parameters for Section 4.7.2

The motion and observation models that were used are (4.6) and (4.7) appropriately, where $h(\cdot)$ is given by the pinhole camera model, and the zero mean Gaussian noise is stated above.

PLANNING USING KITTI DATASET

Our proposed approach for RUB Inference, leverages calculations made in the precursory planning phase to update inference more efficiently. KITTI is a pre-recorded dataset with a single action sequence, i.e. the "future" actions of the robot are pre-determined. Nevertheless, we can still evaluate our approach by appropriately simulating the calculations that would be performed within BSP for that specific (and chosen) single action sequence. In other words, BSP involves belief propagation and objective function evaluations for different candidate actions, followed by identifying the best action via Eq. (4.12) and its execution.

In our case, the performed actions over time are readily available; hence, we only focus on the corresponding future beliefs for such actions given the partial information available to the robot at planning time. Specifically, at each time instant k , we construct the future belief $b[X_{k+1|k}]$ via Eq. (4.10) using the supplied visual odometry as motion model and future landmark observations. Future landmark observations are generated by considering only landmarks projected within the camera field

of view using MAP estimates for landmark positions and camera pose from the propagated belief $b[X_{k+1|k}]$. As in this work the planning phase considers only the already-mapped landmarks, without reasoning about expected new landmarks, each new landmark observation in inference would essentially mean facing inconsistent DA.

To conclude, planning using the KITTI dataset is simulated over a single action in the following manner: current belief is propagated with future action, future measurements are generated by considering already-mapped landmarks, and future belief is solved. Since the "optimal" action is pre-determined by the KITTI dataset there is no need for an objective function evaluation.

PERCEPTION USING KITTI DATASET

After executing the next action, the robot receives a corresponding raw image from the KITTI dataset. The image is being processed through a standard vision pipeline, which produces features with corresponding descriptors [48]. Landmark triangulation is being made after the same feature has been observed at least twice, while following different standard conditions designed to filter outliers. Once a feature is triangulated, it is considered as a landmark, and is added as a new state to the belief. Note that the robot has access only to its current joint belief, consisting of the estimated landmark locations, and the robot past and present pose estimations. Once the observation factors (4.7) are added to the belief, the inference update is being made in two different and separate ways. The first, used for comparison, follows the standard Bayesian inference, by using the efficient methodologies of iSAM2 in order to update inference. The belief of the preceding inference $b[X_{k|k}]$ is being updated with the new motion $\mathbb{P}(x_{k+1}|x_k, u_k)$ and observation factors $\prod_{j \in \mathcal{M}_{k+1|k+1}} \mathbb{P}(z_{k+1}^j | x_{k+1}, l_j)$, thus obtaining $b[X_{k+1|k+1}]$.

The second method follows our proposed paradigm for RUB Inference. The belief from the preceding planning phase, $b[X_{k+1|k}]$, which corresponds to $u_{k|k+1}$ (see (4.26)), is updated with the new measurements. This update is done using UD-OTM-OO which consists of two stages, first using our DA update method (Section 4.6) which updates the predicted DA to the actual DA, followed by the OTM-OO method (Section 4.4.2) which updates measurement values.

RESULTS - KITTI DATASET

The robot travels 1400 steps in the unknown streets of Karlsruhe Germany, while relying only on a monocular camera for localization and mapping and without encountering any substantial loop closures. Differently than Section 4.5 and Section 4.7.1, where the landmarks were omnidirectional and therefore can be spotted from every angle as long as they were in sensing range, when using real world data the angle from which we see a landmark would have crucial affect on data associa-

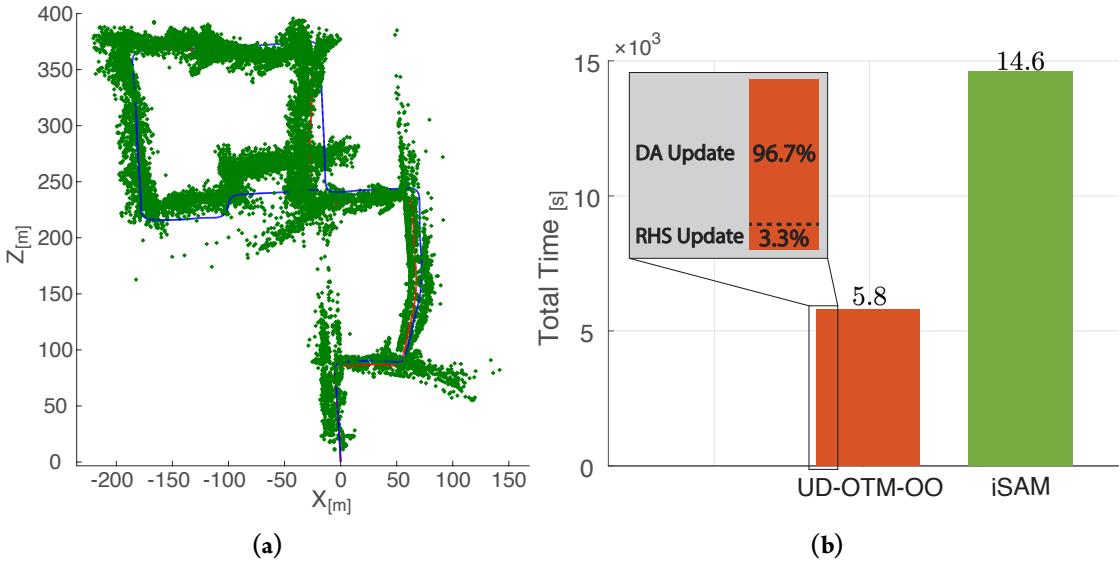


Figure 4.7.3: Experiment layout and results: (a) The city of Karlsruhe, Germany, provided by the KITTI dataset. The robot ground truth is denoted in blue, the estimated trajectory denoted in dotted red line and the estimated landmark locations are denoted in green. (b) Total average running time of inference update for each method, when 100% of the steps were with inconsistent DA.

tion. Figure 4.7.3a presents the ground truth of the robot’s trajectory in blue, the estimated robot’s trajectory in dotted red and the estimated location of observed landmarks in green. Both iSAM and UD-OTM-OO produce the same estimation; therefore, the dotted red-line as well as the green marks represents both iSAM and UD-OTM-OO estimations.

Figure 4.7.3b presents the total computation time of inference update throughout the experiment, for both iSAM, and UD-OTM-OO. The importance of real-world data can be easily noticed by comparing Figures 4.7.3b and 4.7.1b. While the RHS update portion of UD-OTM-OO secured its advantage of two orders of magnitude over iSAM, it is not the case with UD-OTM-OO as a whole. Although for real-world data, UD-OTM-OO is still faster than iSAM, the difference has decreased from order of magnitude in Figure 4.7.1b, to less than half the computation time in Figure 4.7.3b. Since the same machine has been used in both cases, the difference must originate from the data itself. As will be seen later in Figure 4.7.5b, the number of measurements per step is substantially higher when using the real-world data, as well as the occurrences of inconsistent DA. It is worth stressing that iSAM implementation for inference update is C++ based, while UD-OTM-OO implementation consists of a mixture of MATLAB based and C++ based implementation, so under the use of the same platform the computation time difference is expected to be higher.

We continue by discussing the estimation difference, between iSAM and our method UD-OTM-OO. Although our method is algebraically equivalent to estimation via iSAM, for the reader’s assurance we also provide estimation error comparison for both mean and covariance. Despite the algebraic equivalence, we expect to obtain small error values, related to numerical noise, which are

different from absolute zero. The estimation comparison results are presented in Figure 4.7.4: the

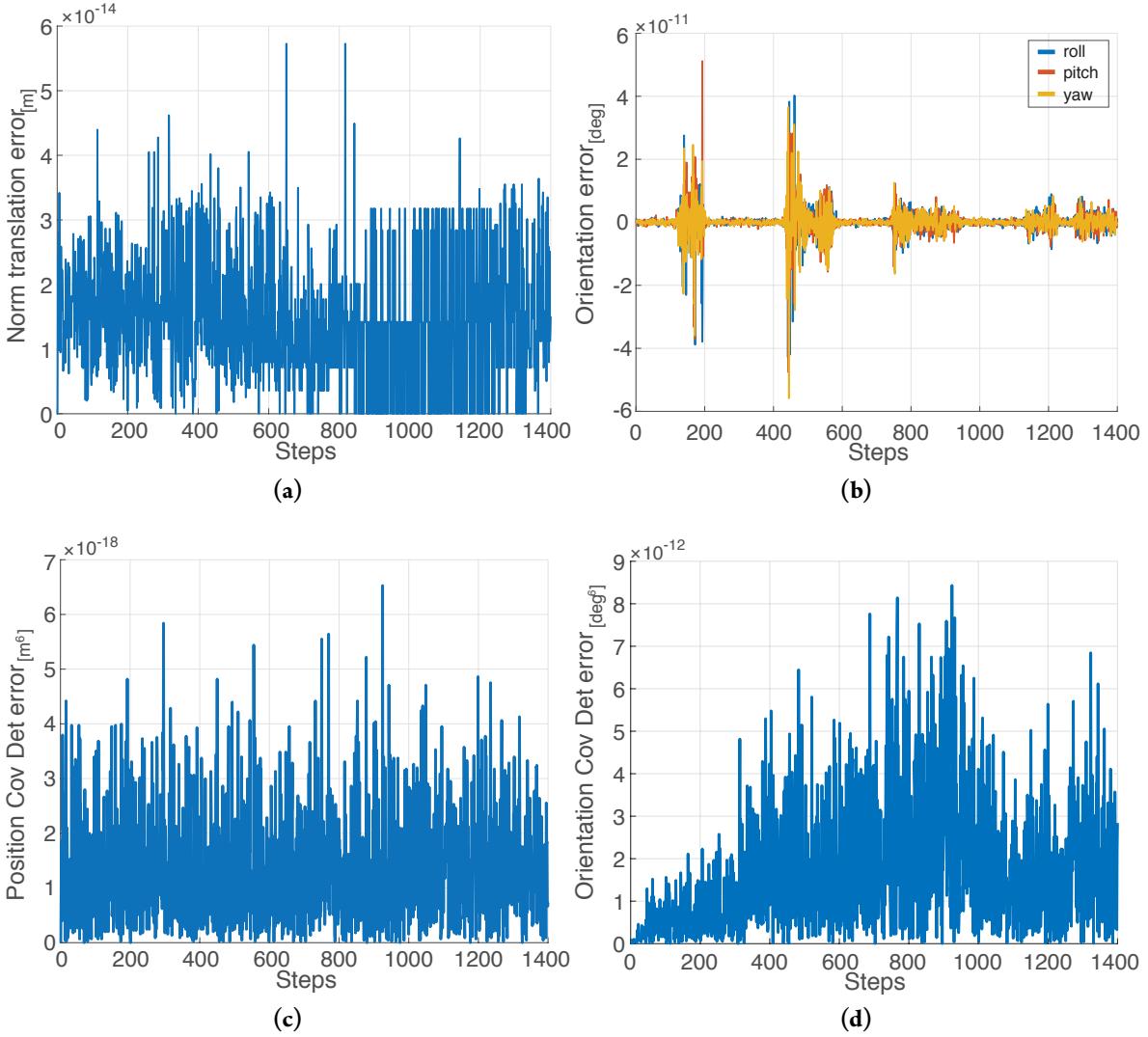


Figure 4.7.4: Relative estimation error between iSAM and UD-OTM-OO, for KITTI dataset experiment (a) Relative translation error, calculated by taking the norm of the difference between the two translation vectors (b) Relative rotation error, calculated by taking the norm of the difference between the two orientation vectors, i.e. Euler angles (c) Relative position covariance error, calculated by taking the determinant of the difference between the two covariance matrices (d) Relative orientation covariance error, calculated by taking the determinant of the difference between the two covariance matrices.

translation mean in Figure 4.7.4a, the mean rotation of the robot in Figure 4.7.4b and the corresponding covariances in Figures 4.7.4c and 4.7.4d accordingly. The mean translation error is calculated by taking the norm of the difference between the two mean translation vectors. The mean rotation error is calculated by taking the norm of the difference between each of the mean body angles. The covariance error is calculated by taking the norm of the difference between the covariance determinants. As can be seen in Figure 4.7.4, the error has a noise like behavior, with values of 10^{-14} for translation

mean, 10^{-11} for mean rotation angles, 10^{-3} for translation covariance and 10^{-2} for rotation angles covariance. For all practical purposes, these values point to a negligible accuracy difference between the two methods.

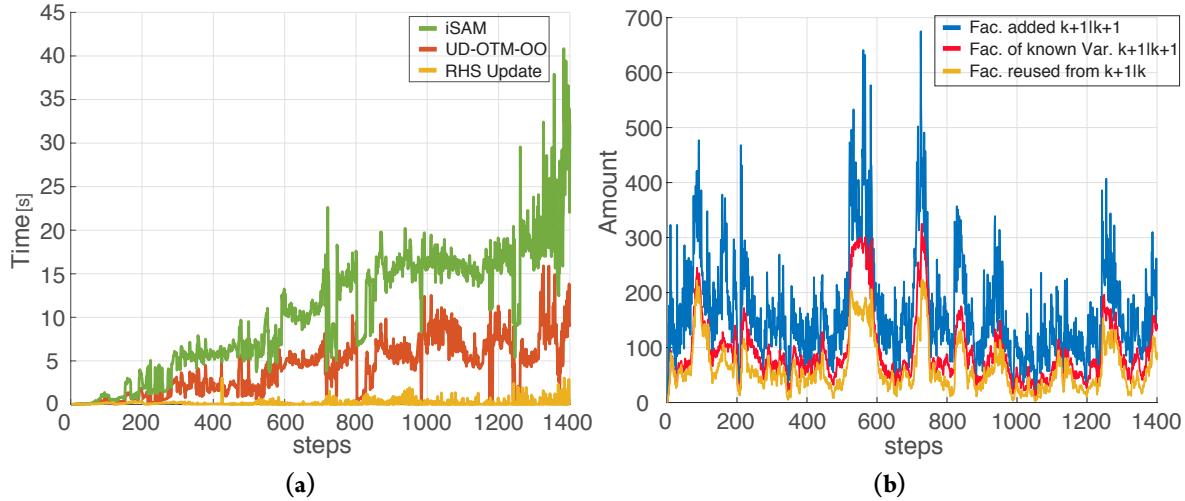


Figure 4.7.5: Per-step analysis of computation time and added factors amount. (a) Inference update computation time per-step comparison between: iSAM - traditional Bayesian inference and UD-OTM-OO - inference update using belief from precursory planning. For reference the RHS update portion out of UD-OTM-OO is denoted in yellow. (b) Number of added factors per step. Number of all factors added in iSAM during inference at time $k + 1$, denoted in blue. Number of factors added in iSAM during inference at time $k + 1$ and relate to known variables, denoted in red. Number of factors that were originally calculated during planning at time $k + 1|k$ and were added by UD-OTM-OO in inference at time $k + 1$, denoted in yellow.

Figure 4.7.5a presents the per-step computation time for inference update of UD-OTM-OO and iSAM, as well as the RHS update portion out of UD-OTM-OO for reference. The RHS Update timing of UD-OTM-OO, denoted by a yellow line, represents the per-step computation time of inference update through RUB inference for consistent DA, i.e. computation time for updating the RHS with the correct measurement values after the DA has been updated. UD-OTM-OO represents the per-step computation time of inference update through RUB inference for the entire process - DA update followed by RHS update. The difference in computational effort between the two, as seen in Figure 4.7.5a, is equivalent to the computation time of the DA update, which represents the need to deal with inconsistent DA between belief from planning $b[X_{k+1|k}]$ and succeeding inference $b[X_{k+1|k+1}]$. The difference in computational effort between UD-OTM-OO and iSAM is attributed to the re-use of calculations made during the precursory planning. This calculation re-use manifests in salvaging factors that have already been considered during the precursory planning.

The reason for the considerable computational time differences between UD-OTM-OO and iSAM is better understood when comparing the factors involved in the computations of each method.

Figure 4.7.5b presents the sum of added factors per-step. In blue, the sum of all factors added at

time $k + 1|k + 1$, as part of standard Bayesian inference update. In red, the portion of the aforementioned factors that relate to states which are already part of the belief $b[X_{k|k}]$. In yellow, the amount of factors added in time $k + 1$ as part of UD-OTM-OO and are shared by both beliefs, $b[X_{k+1|k}]$ and $b[X_{k+1|k+1}]$, i.e. the amount of factors that were originally calculated in the precursory planning time, and were reused by UD-OTM-OO. It is worth stressing the noticeable difference between the number of measurements per step in Figure 4.7.5b when compared to Figure 4.7.2c. The former is exceeding the latter by an order of magnitude.

The difference between the yellow and blue lines in Figure 4.7.5b represents the amount of factors "missing" from the belief $b[X_{k+1|k}]$ in order to match $b[X_{k+1|k+1}]$ (see Section 4.6.2), e.g. for step 725 only 142 have been reused (yellow line) while 675 were eventually added (blue line), leaving 533 new factors to be added during the DA update phase of UD-OTM-OO. This difference can be divided into factors containing only existing states and factors containing new states. Since the red line represents all factors of existing states, the difference between the red and blue lines represents all factors containing new states per time step, e.g. for step 725, out of the 675 factors added during inference (blue line), only 236 are related to previously known states (red line). As mentioned earlier in Section 4.7.2, in this experiment the prediction of future factors does not involve new states, apart from the next future pose(s). For that reason, the amount of factors added during planning has an upper bound represented by the red line, e.g. for step 725, the maximum number of factors that could have been utilized from precursory planning is 236 (red line). Future work can consider a prediction mechanism for new states, such work would set the upper bound somewhere between the red and blue lines.

The difference between the yellow and red lines, both related to factors of existing states, is attributed to the prediction accuracy of the planning stage. Since the factors represented by the red line are already part of the belief in planning time k , a perfect prediction mechanism would have added them all to the belief $b[X_{k+1|k}]$, e.g. for step 725, while there are 236 factors related to previously known states (red line), planning predicted only 142 of them (yellow line). Since the prediction is inherently imperfect (see Section 4.3.2), there would always be some difference between the two. Reducing the gap between the red and yellow lines is a function of the prediction mechanism, while closing the gap further up to the blue line is a function of predicting new variables during planning.

After better understanding the meaning of Figure 4.7.5b, comparing the two graphs in Figure 4.7.5, reveals the connection between the added factors and the computation time, demonstrated by comparing steps 725 and 803 across the aforementioned. For time step 725, we have 675 new factors added in iSAM at inference, while only 142 factors that have been reused by UD-OTM-OO, this difference resulted in inference update running time of 4.2[s] to UD-OTM-OO and 6.1[s] to iSAM. For time step

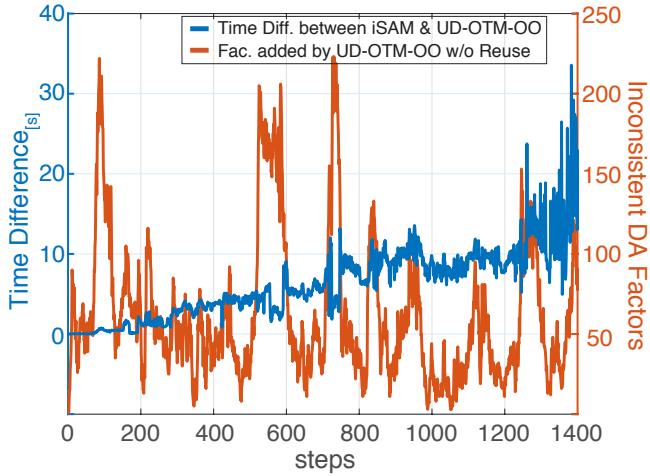


Figure 4.7.6: Inference update computation time analysis between iSAM and UD-OTM-OO. The left vertical blue axis, represents the inference update computation time difference between iSAM and UD-OTM-OO, where positive values suggest $t_{\text{iSAM}} > t_{\text{UD-OTM-OO}}$. The right vertical orange axis, represents the number of factors per step added by UD-OTM-OO that were *not* reused from planning.

803, we have 145 new factors added in iSAM at inference, while only 33 factors that have been reused by UD-OTM-OO, this difference resulted in inference update running time of $0.33[\text{s}]$ to UD-OTM-OO and $6.9[\text{s}]$ to iSAM. Although 6825 new landmarks were added to the state vector between steps 725 and 803 (calculated by the cumulative difference between the blue and red lines between steps 725 and 803), the time difference between UD-OTM-OO and iSAM increased, while UD-OTM-OO running time dropped. This increase in relative running time, in-spite of the substantial growth of the state vector, can be attributed to the drop in the number of factors needed to be added by the DA update phase of UD-OTM-OO.

As anticipated, the larger the gap between $b[X_{k+1|k}]$ and $b[X_{k+1|k+1}]$, i.e. more DA corrections to $b[X_{k+1|k}]$ are required in order to match $b[X_{k+1|k+1}]$, the smaller the computation time difference between RUB inference and standard Bayesian inference, as demonstrated in Figure 4.7.6. The left vertical axis (denoted in blue) presents the computation time difference between iSAM and UD-OTM-OO such that positive values suggest $t_{\text{iSAM}} > t_{\text{UD-OTM-OO}}$. From this blue graph we notice that the time difference between iSAM and UD-OTM-OO is strictly positive and ascending up to fluctuations. While some of these fluctuations can be attributed to machine noise of the measurement process, we provide some explanation for the large time difference drops, i.e. the steps in which the computation difference between iSAM and UD-OTM-OO diminished. The number of factors added by UD-OTM-OO and were not reused from precursory planning are denoted by the orange line in Figure 4.7.6. We can see correlation between large spikes in the number of factors added during the DA update phase of UD-OTM-OO (orange line), and the drops in the time difference between iSAM and UD-OTM-OO computation time difference (blue line), e.g. steps 554 – 591 and

In contrast to previous experiments over synthetic data, we can better see here some dependency over the size of the belief in the UD-OTM-OO method. This dependency seems to be in correlation with that of iSAM2 although less intense, as can be seen by comparing the two methods in Figure 4.7.5a. As in Figure 4.7.2, we can attribute this correlation to the number of re-eliminations performed per step, which are a function of the newly added factors for iSAM and the DA update in UD-OTM-OO (see Section 4.6). As mentioned earlier, in each step UD-OTM-OO encounters inconsistent DA, judging by the difference between the blue and yellow lines in Figure 4.7.5b, each step UD-OTM-OO deals with at least 100 factors that were not reused from planning. Since UD-OTM-OO makes use of iSAM2 methodologies in order to update inconsistent DA, as does iSAM2 to update inference, they share similar computational sensitivities, which manifest in similar computation time trends. This similarity sensitivity is attributed in our opinion to the elimination process required in order to introduce new factors into the belief. Future work for reducing eliminations by anticipating required ordering, would break this dependency and provide additional improvement in computation time as well as in reducing the sensitivity to state dimensionality.

4.8 SOME BROADER PERSPECTIVE

In this section we briefly discuss the motivation for RUB Inference and provide some broader perspective to possible future usage. As mentioned earlier, the RUB Inference paradigm deals with inference update within a plan-act-infer framework. By re-using calculations from the precursory planning session, it offers reduced computation time without affecting estimation accuracy.

Decision making under uncertainty in high dimensional state spaces is computationally intractable, and as such the majority of the plan-act-infer computation time can be ascribed to it. For example, let us consider BSP under the simplified Maximum Likelihood (ML) assumption, with a planning horizon of three lookahead steps and three candidate actions per step. The first level of the belief tree would consist of three future beliefs, one for each candidate action, each of which is propagated with each of the three candidate actions, resulting in nine future beliefs in the second level of the belief tree, and again for the last lookahead step with 27 future beliefs in the third and last level of the belief tree. This would result in total of 39 future beliefs that constitute the belief tree, i.e. 39 belief updates, whereas only a single belief update is required during inference update. In this toy example the computational load of inference update constitutes therefore only 2.5% of the plan-act-infer framework (assuming all belief updates have the same computational load). So why should we bother with the efficiency of the inference update process in the first place?

The answer to this question is twofold, the first part deals with RUB Inference paradigm as a

stand-alone approach for inference update, and the second with its possible contribution to future research.

Although we present RUB Inference as part of a plan-act-infer system, it can also be used in the passive case, i.e. not as part of a plan-act-infer system. Imagine a set of candidate beliefs, calculated offline and stored away for future usage. When in-need to perform belief update, we can search this set of candidate beliefs for the belief closest to last posterior as well as to the newly received information. Once we locate this closest belief, we use RUB Inference to update it to match current information thus saving computational load without affecting accuracy. The reason we consider in this work RUB Inference as part of a plan-act-infer system, lies within the problem of locating the closest candidate belief. By using beliefs from precursory planning as candidates we have a small set of candidates to look through, moreover we can ensure that in the worst case scenario (i.e. all predictions from precursory planning are wrong) we would match Bayesian inference performance, thus averting from the complicated problem of searching within belief space. For the general case of having a set of previously calculated beliefs, used as candidates for re-use under RUB Inference, one would need to deal with few issues, some of which are: how to store the beliefs to facilitate an efficient search, how to efficiently search the set of candidate beliefs, what high-dimensional belief-distance to use, how to interpret belief-distance into computational load i.e. what will be considered as close enough. It goes without saying that the computational load of locating the closest belief should be small enough for RUB Inference to still have a computational advantage.

Secondly, the paradigm shift suggested by RUB Inference provides a pathway to new and exciting research directions. For example, RUB Inference is a key building block in JIP (Chapter 3), first presented in [17] and later in [19], which strives to create a unified framework that deals with both inference and BSP under the same governing system, thus allowing to maximize the calculation re-use potential available in both inference and planning.

While RUB Inference provides efficient inference update, it is also relied upon to facilitate calculation re-use between different planning sessions (see Chapter 6). As part of BSP, it is required to create a belief tree, as deep as the planning horizon, where each node in the said tree represents a future belief with specific candidate actions and future measurements. In Chapter 6 we make use of RUB Inference paradigm in order to incrementally update the belief tree as part of planning, thus saving valuable computation time from the planning stage.

Consequently, apart from being a more computationally efficient approach for inference update, the paradigm shift suggested by RUB Inference provides a basis for exciting new research, e.g. it is the building block that facilitates the JIP concept as a whole.

4.9 CONCLUDING RUBI

Conventional Bayesian inference updates the belief from a previous time step with new incoming information. In this chapter we introduced an alternative paradigm, utilizing the similarities between inference and planning to efficiently update inference using information from precursory planning phase. Given a future belief from precursory planning and newly acquired data, we appropriately update the former with the latter while taking into consideration data association inconsistencies which might occur. The resulting approach, RUB Inference, saves valuable computation time in inference without affecting the estimation accuracy.

We evaluated our approach in simulation and using real-world data from the KITTI dataset, considering active SLAM as application, and compared it against iSAM2, a state-of-the-art incremental Bayesian inference approach. Results from real-world evaluation indicate that our approach is more efficient computationally by at least a factor of two compared to iSAM2, without affecting the solution accuracy. The improvement magnitude is in direct correlation with the quality of the prediction mechanism being used in planning, meaning a better prediction mechanism would increase the approach's efficiency. A particular appealing aspect of our method, that we demonstrated using synthetic data, is that loop closures computational burden during inference is elevated, thanks to the utilization of similar calculations already made during precursory planning. When loop closures were correctly predicted during the planning phase, our method utilized these calculations instead of recalculating them in inference, resulting in reducing computation time by a factor of two orders of magnitude in the shown results.

This paper suggests a novel general concept for leveraging calculations from the decision making stage for efficient inference update, thus enabling to reduce inference computation time without affecting accuracy. Based on this concept, under the assumption of high-dimensional Gaussian beliefs, we developed approaches based on the square root information matrix, to efficiently update inference. We strongly believe this novel concept is applicable for more general distributions in any autonomous system involving both inference and decision making under uncertainty. Based on our findings, we strongly believe this paradigm shift opens new research directions and can be further extended in various ways, e.g. iX-BSP- incremental expectation BSP (Chapter 6) leverages RUB Inference to reuse calculations across different planning sessions.

*You have to learn the rules of the game.
And then you have to play better than anyone else.*

Sen. Dianne Fienstein

5

Introducing iX-BSP

IN THIS CHAPTER WE INTRODUCE INCREMENTAL EXPECTATION BSP, OR iX-BSP, WHICH INCREMENTALLY UPDATES THE EXPECTATION RELATED CALCULATIONS IN X-BSP BY RE-USING PREVIOUS PLANNING SESSIONS. We suggest reformulating the original intractable problem of X-BSP to account for calculation re-use, thus improving calculation time without affecting the solution accuracy. This chapter is organized as follows. Section 5.1 following Chapter 2, provides an in-depth review of iX-BSP related work. Section 5.2 introduces the concept of iX-BSP and reviews the contributions of this chapter. Section 5.3 reviews current formulation of a plan-act-infer system, focusing on decision making under uncertainty. Section 5.5 ponders about the expectation in the X-BSP problem and the common Maximum Likelihood (ML) approximation and its repercussions.

5.1 RELATED WORK

As mentioned in Chapter 2, in strike contrast to the vast amount of research invested in approximating the X-BSP problem, only few tried re-using calculations, this section shortly reviews some of these efforts to provide the reader with a better understanding of iX-BSP specific related work.

POMCP [64] is the POMDP generalization of MCTS [9]. POMCP is a point-based POMDP;

as such, it is implicitly using particle filtering to express and update future beliefs, thus breaking the curse of dimensionality. It also maintains a history search tree, containing all possible actions and only sampled measurements, thus breaking the curse of history. At each planning session POMCP develops the tree using a black box simulator and search for the optimal policy through maintaining an upper bound over each tree node using the PO-UCT algorithm (see section 3.1 in [64]). Once an action has been executed, POMCP is pruning the existing tree of all branches but the one containing the current history, thus re-using the statistics obtained so far for the appropriate segment of the tree. Other than parameter tuning, POMCP does not require any offline calculations, although it does require the existence of a black box simulator for sampling future scenarios and their rewards.

Adaptive Belief Tree (ABT) [46] builds on POMCP but thrives to reuse an existing policy instead of calculating a new one from scratch. Given some offline calculated policy, ABT resamples only the parts affected by newly acquired information, thus re-using the rest of the policy. In addition to offline policy calculation, ABT also requires as an input the set of all states affected by the newly acquired information, which is considered as given. Although re-using unaffected parts of the policy, ABT substitutes the affected parts of the policy with freshly sampled rather than incrementally updating them with the new information.

DESPOT [76] is also a point-based POMDP, which implicitly uses particle filtering to express and update the belief. Same as POMCP it also maintains a history search tree containing all possible actions and only sampled measurements. Similarly to POMCP, at each planning session DESPOT incrementally develops the tree using a simulator while searching for the optimal policy. Unlike POMCP, DESPOT makes use of upper and lower bounds over an estimate of the regularized weighted return value of each tree node, which is supposed to be less greedy than the use of PO-UCT. The lower bound is calculated over the user defined default policy (calculated offline). Once a policy has been chosen, the DESPOT tree is pruned of all other actions; this action-pruned DESPOT is denoted as the DESPOT policy. While following the DESPOT policy, if the agent encounters a scenario not present in the DESPOT policy it will follow the default policy from then on. Similarly it follows the default policy when encountering a leaf node of DESPOT policy. DESPOT also makes use of regularization in order to assure the desired performance bound while evading from potentially overfitting due to the sampled scenarios. The regularization constant as well as other heuristic related parameters are calculated offline and are case sensitive. As opposed to POMPC which operates under an MPC setting, DESPOT follows a DESPOT policy to the end of its horizon and does not make any re-use of the statistical results.

is-DESPOT [50] builds on DESPOT [76] and introduces importance sampling into the scenario sampling procedure. Under is-DESPOT, scenarios are sampled from some importance distribution. The importance sampling distribution is calculated offline and is also case sensitive without explicitly

addressing the deployment problem.

DESPOT- α [20] also builds on DESPOT [76] and introduces α vector approximation to the implicit particle filtering problem of propagating and maintaining future beliefs, thus better representing belief uncertainty under relatively large measurement space.

In contrast to the aforementioned efforts, although under some more simplifying assumptions amongst them ML, both [7] and [42] re-use computationally expensive calculations during planning. In their work, Chaves and Eustice [7], consider a Gaussian belief under ML-BSP in a Bayes tree [35] representation. All candidate action sequences consider a shared location (entrance pose), thus enabling to re-use a lot of the calculations through state ordering constraints. That work enables to efficiently evaluate a single candidate action across multiple time steps, and is conceptually applicable to multiple candidate actions at a single time step. While Kopitkov and Indelman [42, 43], also consider a Gaussian belief under ML-BSP, they utilize a factor graph representation of the belief while considering an information theoretic cost. Using an (augmented) determinant lemma, they are able to avert from belief propagation while re-using calculations throughout the planning session. Although they consider calculation re-use within the same planning session, their work can be augmented to consider re-use also between planning sessions.

To the best of our knowledge, in spite of the aforementioned research efforts, calculation re-use has only been done over ML-BSP, with restricting assumptions. While ML-BSP is widely used, the pruning of X-BSP by considering only the most likely measurements, might mean choosing a sub-optimal action in case the biggest available reward is not the most likely one, in particular in presence of significant estimation uncertainty. Although under some conditions few of the aforementioned approaches re-use previously calculated policies, and even selectively re-calculate parts of the policy to account for newly acquired information, they do not address the problem of identifying the parts of the policy affected by the new information nor they incrementally update the policy to account for it. As for today, up to our work, X-BSP approaches do not re-use calculations across planning sessions.

5.2 iX-BSP CONTRIBUTIONS

While the aforementioned research efforts mainly focused on approximating the X-BSP, we suggest a new paradigm, challenging the standard formulation of X-BSP by re-using calculations across planning sessions thus saving valuable computation time not at the expense of accuracy.

In contrast to both POMPC and DESPOT which are point-based approximations of X-BSP, iX-BSP is identical to the general intractable formulation of X-BSP with the single exception of calculation re-use. iX-BSP requires no offline calculations and does not rely on any case sensitive



Figure 5.2.1: Illustrating the difference between iX-BSP and X-BSP using a simple decision making problem. Veronica would like to cross the park with her son as quickly as possible. (a) Before entering the park Veronica, knowing the layout, considers all possible routs, denoted by colored arrows. (b) After entering the park Veronica obtains new information which renders her existing plan suboptimal. Under X-BSP Veronica would replan from scratch while under iX-BSP she would simply incrementally update her existing plan with the new information.

heuristics. Instead of calculating the expected belief-dependent rewards, which form the objective value, from scratch, iX-BSP selectively re-uses previously sampled measurements along with their associated beliefs. While POMPC re-uses previously calculated statistics between planning sessions of the pruned search tree it disregards impact of the newly received information over the sampled scenarios. Although ABT accounts for this new information by re-calculating affected parts of the policy, it does not address how to locate the affected segments of the policy or how to incrementally update them rather than discard them. On the other hand, iX-BSP incrementally updates the re-used beliefs with the updated posterior information, using our previous work on efficient belief update under RUB Inference (see Chapter 4).

Selectively resampling measurements as part of iX-BSP, potentially results in estimating the expected objective value through multiple different distributions. We identify this problem within the Multiple Importance Sampling (MIS) problem [74], residing in the field of importance sampling [24] and formulate iX-BSP accordingly. Figure 5.2.1 illustrates the advantage of iX-BSP over X-BSP through a simple decision making under uncertainty problem, in-which Veronica and her son would like to cross the park in the fastest possible route. Assuming Veronica is already familiar with the park layout, she could solve her decision making problem by considering all four possible routes through the park (denoted by green, red, blue and black arrows in Figure 5.2.1a). As a result, Veronica chooses the blue path and enters the park only to find it populated as illustrated in Figure 5.2.1b. This newly acquired information changes the optimality and even validity of Veronica’s solution. Un-

der X-BSP Veronica would calculate a new plan from scratch; instead, under iX-BSP Veronica can simply update her existing plan with the newly acquired information, resulting with picking the black path as the fastest one.

While Veronica’s planning problem in Figure 5.2.1 provides some intuition on how iX-BSP provides the same solution as X-BSP but faster, in real life scenarios the desirable accuracy sometime lies within the bounds of a suboptimal solution. As mentioned, iX-BSP is not an approximation of X-BSP, it updates some precursory planning tree with all of the new posterior information.

Additional contribution of our work comes to satisfy the desire to controllably sacrifice accuracy for performance, introducing the wildfire approximation. While iX-BSP updates some existing planning tree to exactly match current information, wildfire introduces a concept of “close enough” defined by a wildfire distance threshold. Under iX-BSP with wildfire, whenever a belief of the existing planning tree is “close enough” to its updated counterpart, this “close enough” belief is considered as already updated, along with all of its decedents in the planning tree. The choice of wildfire threshold value would directly affect the obtainable objective value. When the threshold is taken to its minimum, i.e. zero distance, there is no approximation and the obtainable objective value would statistically match the one obtained by iX-BSP without wildfire. As we increase the allowable wildfire distance threshold, it is as if we consider some or even all of the newly acquired information as irrelevant, which would directly impact the obtainable objective value. In this work we formulate the affect of the wildfire threshold over the obtainable objective value, and support it with both analytical proof and empirical results. Moreover, we provide results indicating considerable reduction in computation time under the use of wildfire in iX-BSP.

As iX-BSP is formulated over the original un-approximated problem of X-BSP, we go further and support our claim made in [18], that iX-BSP can be utilized to also reduce valuable computation time of existing approximations of X-BSP. Considering the commonly used approximation ML-BSP, we formulate incremental ML-BSP, referred to as iML-BSP, by simply enforcing the ML assumption over iX-BSP, as being done over X-BSP. Under ML-BSP, beliefs are propagated with zero innovation by considering just the most likely measurement for each candidate action, thus averting from expectation and minimizing the *curse of history*. Given access to calculations from precursory planning, at each look ahead step i in the current planning session, iML-BSP considers the appropriate sample from the i_{th} look ahead step in the precursory planning session for re-use. If the sample constitutes an adequate representation, of the measurement likelihood we would have considered at the i_{th} look ahead step in current planning session, then iML-BSP utilizes the associated previously solved belief from the precursory planning session. If the mentioned sample is considered as an inadequate representation of the mentioned measurement likelihood, iX-BSP follows the course of ML-BSP, and the most likely measurement of the nominal measurement likelihood is

considered instead.

To summarize, the contributions of our work on iX-BSP are as follows: (a) We present a novel paradigm for incremental expectation belief space planning with selective resampling (iX-BSP). Our approach incrementally calculates the expectation over future observations by a set of samples comprising of newly sampled measurements and re-used samples generated at different planning sessions. (b) We identify the problem of iX-BSP with selective resampling as a Multiple Importance sampling problem, and provide the proper formulation while considering the balance heuristic. (c) We evaluate iX-BSP in simulation and provide statistical comparison to X-BSP, which calculates expectation from scratch, while considering the problem of autonomous navigation in unknown environments, across different randomized scenarios. (d) We introduce the wildfire approximation into iX-BSP, which enables one to controllably trade accuracy for performance. (e) We provide an analytical proof of the affect the choice of a wildfire threshold would have over the objective value, in the form of bounds over the objective value error. (f) We provide empirical results of using wildfire within iX-BSP, as well as the affect wildfire holds over the objective value error. (g) We support our claim, that iX-BSP can be used to improve approximations of the general problem of X-BSP, by introducing to iX-BSP the commonly used ML approximation, and denote it as iML-BSP. The novel approach of iML-BSP, incrementally calculates the expectation over future observations, while considering either the most likely observation or some previously sampled observation, given from a precursory planning session. (h) We evaluate iML-BSP in simulation as well as in real-world experiments and compare it to the commonly used approximation for the X-BSP problem, ML-BSP, while considering the problem of autonomous navigation in unknown environments and active visual-SLAM setting with belief over high dimensional state space.

5.3 BSP TODAY

This section provides the theoretical background for belief space planning (BSP), starting with recapping belief definition, followed by the BSP formulation and the common Maximum Likelihood (ML) approximation. While the formulation, as well as the suggested paradigm, are impartial to a specific belief distribution, throughout this chapter we also provide the conventional case which deals with Gaussian distributions.

5.3.1 BELIEF DEFINITION

Although already covered in Section 4.2, for the reader’s convenience we shortly recap the belief definition. Let x_t denote the agent’s state at time instant t and \mathcal{L} represent the mapped environment thus

far. The joint state, up to and including time k , is defined as $X_k = \{x_o, \dots, x_k, \mathcal{L}\}$. We shall be using the notation $t|k$ to refer to some time instant t while considering information up to and including time k . The unique time notation is required since this paper makes use of both current and future time indices in the same equations. Let $z_{t|k}$ and $u_{t|k}$ denote, respectively, measurements and the control action at time t , while the current time is k . The measurements and controls up to time t given current time is k , are represented by

$$z_{1:t|k} \doteq \{z_{1|k}, \dots, z_{t|k}\}, u_{o:t-1|k} \doteq \{u_{o|k}, \dots, u_{t-1|k}\}, \quad (5.1)$$

The posterior probability density function (pdf) over the joint state, denoted as the *belief*, is given by

$$b[X_{t|k}] \doteq \mathbb{P}(X_t | z_{1:t|k}, u_{o:t-1|k}) = \mathbb{P}(X_t | H_{t|k}), \quad (5.2)$$

where $H_{t|k} \doteq \{u_{o:t-1|k}, z_{1:t|k}\}$ represents history at time t given current time k . The propagated belief at time t , i.e. belief $b[X_{t|k}]$ lacking the measurements of time t , is denoted by

$$b^-[X_{t|k}] \doteq b[X_{t-1|k}] \cdot \mathbb{P}(x_t | x_{t-1}, u_{t-1|k}) = \mathbb{P}(X_t | H_{t|k}^-), \quad (5.3)$$

where $H_{t|k}^- \doteq H_{t-1|k} \cup \{u_{t-1|k}\}$. Using Bayes rule, Eq. (5.2) can be rewritten as

$$b[X_{t|k}] \propto \mathbb{P}(X_o) \prod_{i=1}^t \left[\mathbb{P}(x_i | x_{i-1}, u_{i-1|k}) \prod_{j \in \mathcal{M}_{i|k}} \mathbb{P}(z_{i,j|k} | x_i, l_j) \right], \quad (5.4)$$

where $\mathbb{P}(X_o)$ is the prior on the initial joint state, and $\mathbb{P}(x_i | x_{i-1}, u_{i-1|k})$ and $\mathbb{P}(z_{i,j|k} | x_i, l_j)$ denote, respectively, the motion and measurement likelihood models. Here, $z_{i,j|k}$ represents an observation of landmark l_j from robot pose x_i , while the set $\mathcal{M}_{i|k}$ contains all landmark indices observed at time i , i.e. it denotes data association (DA). The DA of a few time steps is denoted by $\mathcal{M}_{1:i|k} \doteq \{\mathcal{M}_{1|k}, \dots, \mathcal{M}_{i|k}\}$.

5.3.2 BELIEF SPACE PLANNING FORMULATION

Although shortly mentioned in Section 4.2.2 as part of RUB Inference theoretical background, we provide here a more thorough formulation of the belief space planning problem as required for iX-BSP.

The purpose of BSP is to determine an optimal action given an objective function J , belief $b[X_{k|k}]$ at planning time instant k and, considering a discrete action space, a set of candidate actions \mathcal{U}_k . While

these actions can be with different planning horizons, we consider for simplicity the same horizon of L look ahead steps for all actions, i.e. $\mathcal{U}_k = \{u_{k:k+L-1}\}$.

The optimal action for planning at time k for horizon of L look-ahead steps is given by

$$u_{k:k+L-1|k}^* = \arg \max_{u_{k:k+L-1|k} \in \mathcal{U}_k} J(u_{k:k+L-1|k}), \quad (5.5)$$

where the general objective function $J(\cdot)$ is defined as

$$J(u) \doteq \mathbb{E}_{z_{k+1:k+L|k}} \left[\sum_{i=k+1}^{k+L} r_i(b[X_{i|k}], u_{i-1|k}) \right], \quad (5.6)$$

with $u \doteq u_{k:k+L-1|k}$, immediate rewards (or costs) r_i and where the expectation is with respect to future observations $z_{k+1:k+L|k}$ while,

$$z_{k+1:k+L|k} \sim \mathbb{P}(z_{k+1:k+L|k} | H_{k|k}, u_{k:k+L-1}). \quad (5.7)$$

The expectation in (5.6) can be written explicitly

$$J(u) = \int_{z_{k+1|k}} \mathbb{P}(z_{k+1|k} | H_{k|k}, u_{k|k}) \cdot r_{k+1}(\cdot) + \dots + \int_{z_{k+1:i|k}} \mathbb{P}(z_{k+1:i|k} | H_{k|k}, u_{k:i-1|k}) \cdot r_i(\cdot) + \dots \quad (5.8)$$

Using the chain rule and the Markov assumption, we can re-formulate the joint measurement likelihood (5.7), as

$$\mathbb{P}(z_{k+1:k+L|k} | H_{k|k}, u_{k:k+L-1}) = \prod_{i=k+1}^{k+L} \mathbb{P}(z_{i|k} | H_{i|k}^-) \quad (5.9)$$

where $H_{i|k}^-$ is a function of a specific sequence of measurement realization, i.e.

$$H_{i|k}^- = H_{k|k} \cup \{z_{k+1:i-1|k}, u_{k:i-1|k}\}. \quad (5.10)$$

Using (5.9), we can re-formulate (5.8) as

$$J(u) = \int_{z_{k+1|k}} \mathbb{P}(z_{k+1|k} | H_{k+1|k}^-) \left[r_{k+1}(b[X_{k+1|k}], u_{k|k}) + \dots + \int_{z_{i|k}} \mathbb{P}(z_{i|k} | H_{i|k}^-) [r_i(b[X_{i|k}], u_{i-1|k}) + \dots] \right], \quad (5.11)$$

where each integral accounts for all possible measurement realizations from an appropriate look ahead

step, with $i \in (k+1, k+L]$ and $b[X_{i|k}] = \mathbb{P}(X_{i|k}|H_{i|k}^-)$.

Evaluating the objective for each candidate action in \mathcal{U}_k involves calculating (5.11), considering all different measurement realizations. As solving these integrals analytically is typically not feasible, in practice these are approximated by sampling future measurements. Although the measurement likelihood $\mathbb{P}(z_{i|k}|H_{i|k}^-)$ is unattainable, one can still sample from it. Specifically, consider the i -th future step and the corresponding $H_{i|k}^-$ to some realization of measurements from the previous steps. In order to sample from $\mathbb{P}(z_{i|k}|H_{i|k}^-)$, we should marginalize over the future robot pose x_i and landmarks \mathcal{L}

$$\mathbb{P}(z_{i|k}|H_{i|k}^-) = \int_{x_i} \int_{\mathcal{L}} \mathbb{P}(z_{i|k}|x_i, \mathcal{L}) \cdot \mathbb{P}(x_i, \mathcal{L}|H_{i|k}^-) dx_i d\mathcal{L}, \quad (5.12)$$

where $\mathbb{P}(x_i, \mathcal{L}|H_{i|k}^-)$ can be calculated from the belief $b^-[X_{i|k}] \doteq \mathbb{P}(X_{i|k}|H_{i|k}^-)$. We approximate the above integral via sampling as summarized in Alg. 2. One can also choose to approximate further by considering only landmark estimates $\hat{\mathcal{L}}$ (i.e. without sampling \mathcal{L}).

Algorithm 2 Sampling $z_{i|k} \sim \mathbb{P}(z_{i|k}|H_{i|k}^-)$

-
- 1: $\chi_i \doteq \{x_i, \mathcal{L}\} \sim \mathbb{P}(x_i, \mathcal{L}|H_{i|k}^-)$
 - 2: Determine data association $\mathcal{M}_{i|k}(x_i, \mathcal{L})$
 - 3: $z_{i|k} = \{z_{i,j|k}\}_{j \in \mathcal{M}_{i|k}(\chi_i)}$ with $z_{i,j|k} \sim \mathbb{P}(z_{i,j|k}|x_i, l_j)$
 - 4: **return** $z_{i|k}$ and χ_i
-

Each sample χ_i and the determined DA (lines 1-2 of Alg. 2) define a measurement likelihood $\mathbb{P}(z_{i|k}|\chi_i, \mathcal{M}_{i|k}(\chi_i)) = \prod_{j \in \mathcal{M}_{i|k}(\chi_i)} \mathbb{P}(z_{i,j|k}|x_i, l_j)$ from which observations are sampled in line 3. Considering n_x samples, $\{\chi_i^n\}_{n=1}^{n_x}$, we can approximate Eq. (5.12) by

$$\mathbb{P}(z_{i|k}|H_{i|k}^-) \approx \eta_i \sum_{n=1}^{n_x} \omega_i^n \cdot \mathbb{P}(z_{i|k}|\chi_i^n, \mathcal{M}_{i|k}(\chi_i^n)), \quad (5.13)$$

where ω_i^n represents the n -th sample weight, χ_i^n , and $\eta_i^{-1} \doteq \sum_{n=1}^{n_x} \omega_i^n$. Here, since all samples are generated from their original distribution (corresponding to the proposal distribution in importance sampling), see line 1, we have identical weights.

For each sample $\chi_i^n \in \{\chi_i^n\}_{n=1}^{n_x}$, we can generally consider n_z measurement samples (line 3), providing the set $\{z_{i|k}^{n,m}\}_{m=1}^{n_z}$. In other words, Alg. 2 yields $n_x \cdot n_z$ sampled measurements, denoted by $\{z_{i|k}\}$, for a given realization of $z_{k+1:i-1|k}$. Thus, considering all such possible realizations, we get $(n_x \cdot n_z)^{i-k}$ sampled measurements for the look ahead step at time instant i , i.e. the $(i-k)$ -th look ahead step for planning time instant k .

We can now write an unbiased estimator for (5.11), considering the $(n_x \cdot n_z)^{i-k}$ sampled measurements. In particular, for the look ahead step at time i , we get

$$\mathbb{E}_{z_{k+1:i|k}} [r_i(b[X_{i|k}], u_{i-1|k})] \approx \eta_{k+1} \sum_{\{z_{k+1|i}\}} \omega_{k+1}^n \left(\dots \left(\eta_i \sum_{\{z_{i|k}\}} \omega_i^n \cdot r_i(b[X_{i|k}], u_{i-1|k}) \right) \dots \right) \quad (5.14)$$

where $H_{i|k}$ varies with each measurement realization. When the measurements that are used to estimate the expectation are being sampled from their nominal distributions, then all weights equal 1, i.e. $\omega_i^n = 1 \forall i, n$, and evidently each normalizer equals the inverse of the sum of samples, i.e. $\eta_i^{-1} = n_x \cdot n_z \forall i$

$$J(u) = \sum_{i=k+1}^{k+L} \left[\frac{1}{(n_x \cdot n_z)^{i-k}} \sum_{\{z_{k+1|i}\}} \dots \sum_{\{z_{i|k}\}} r_i(b[X_{i|k}], u_{i-1|k}) \right]. \quad (5.15)$$

The above exponential complexity makes the described calculations quickly infeasible, due to both curse of dimensionality and history. In practice, approximate approaches, e.g. Monte-Carlo tree search [64], must be used. However, in this work we prefer to present our paradigm considering the above formulation, without any further approximations, referring to it as X-BSP. We believe our proposed concept can be applied in conjunction with existing approximate approaches; in particular, we demonstrate this on the commonly used approximation for the X-BSP problem - the Maximum Likelihood approximation.

5.3.3 BELIEF SPACE PLANNING UNDER ML

A very common approximation to Eq. (5.6) is based on the maximum likelihood (ML) observations assumption (see e.g. [29, 40, 58]). This approximation, referred to as ML-BSP, is often used in BSP and in particular in the context of active SLAM: Instead of accounting for different measurement realizations, only the most likely observation is considered at each look ahead step, which corresponds to $n_x = n_z = 1$ where the single sample is the most likely one. So under ML, the expectation from Eq. (5.6) is omitted, and the new objective formulation is given by

$$J^{ML}(u) \doteq \sum_{i=k+1}^{k+L} r_i(b[X_{i|k}], u_{i-1|k}), \quad (5.16)$$

thus drastically reducing complexity at the expense of sacrificing performance. While the future belief $b[X_{i|k}]$ is given by $\mathbb{P}(X_{0:i}|H_{k|k}, u_{k:i-1}, z_{k+1:i|k}^{ML})$, and for the Gaussian case $z_{k+1:i|k}^{ML}$ are the measurement

model mean-values.

5.4 DEFINING THE iX-BSP PROBLEM

Consider the planning session at time instant k has been solved by evaluating the objective (5.6) via appropriate measurement sampling for each action in \mathcal{U}_k and subsequently choosing the optimal action $u_{k:k+L-1|k}^*$. A subset of this action, $u_{k:k+l-1|k}^* \in u_{k:k+L-1|k}^*$ with $l \in [1, L]$, is now executed, new measurements $z_{k+1:k+l|k+l}$ are obtained and the posterior belief $b[X_{k+l|k+l}]$ in inference is calculated, upon which a new planning session is initiated.

Determining the optimal action sequence at time instant $k + l$ involves evaluating the objective function for each candidate action $u' \doteq u_{k+l:k+l+L-1|k+l} \in \mathcal{U}_{k+l}$

$$J(u') \doteq \mathbb{E} \left[\sum_{i=k+l+1}^{k+l+L} r_i (b[X_{i|k+l}], u'_{i-1|k+l}) \right], \quad (5.17)$$

where the expectation is with respect to future observations $z_{k+l+1:k+l+L|k+l}$. Existing approaches perform these costly evaluations from scratch for each candidate action. Our *key observation* is that expectation related calculations from two successive X-BSP planning sessions at time instances k and $k + l$ are similar and hence can often be re-used. The iX-BSP approach would evaluate the objective function (5.17) more efficiently by appropriately re-using calculations from preceding planning sessions.

At this point, we summarize our assumptions for iX-BSP.

Assumption 1. *Calculations from a previous planning session are accessible from the current planning session.*

Assumption 2. *The planning horizon of current time $k + l$, overlaps the planning horizon of the precursory planning time k , i.e. $l \in [1, L]$.*

Assumption 3. *Action sets \mathcal{U}_{k+l} and \mathcal{U}_k overlap in the sense that actions in \mathcal{U}_k that overlap in the executed portion of the optimal action also partially reside in \mathcal{U}_{k+l} . In other words, $\forall u \in \mathcal{U}_k$ with $u \doteq \{u_{k:k+l-1|k}, u_{k+l:k+L-1|k}\}$ and $u_{k:k+l|k} \equiv u_{k:k+l-1|k}^* \exists u' \in \mathcal{U}_{k+l}$ such that $u' \doteq \{u'_{k+l:k+L-1}, u'_{k+L:k+l+L-1}\}$ and $u'_{k+l:k+L-1} \cap u_{k+l:k+L-1|k} \notin \emptyset$.*

Unlike assumption 1, which is an integral part of iX-BSP, assumptions 2-3 exist only as a mean to create a smaller group of candidate beliefs for re-use. By limiting ourselves to beliefs with a shared history, mostly the same action sequence and of the same future time, we obtain a relatively small set

that is likely to produce a viable candidate for re-use. One can relax these assumptions after addressing the problem of efficiently searching a set of candidate beliefs.

5.5 EXPECTATION AND THE MAXIMUM LIKELIHOOD ASSUMPTION

In this section we provide a glimpse behind the curtains of X-BSP and ML-BSP. As seen in Section 5.3.3, the ML assumption greatly simplify the X-BSP problem by eliminating the expectation over future measurements, as formulated in Eq.(5.16). However this simplification comes with a cost, which we will discuss in this section. By considering only the most likely measurement, rather than taking into account all possible future measurements we risk choosing a sub-optimal action. In order to have a tangible example to discuss and ponder about, we start with the results of a planning session, for which X-BSP and ML-BSP produced different optimal actions. We show the results of

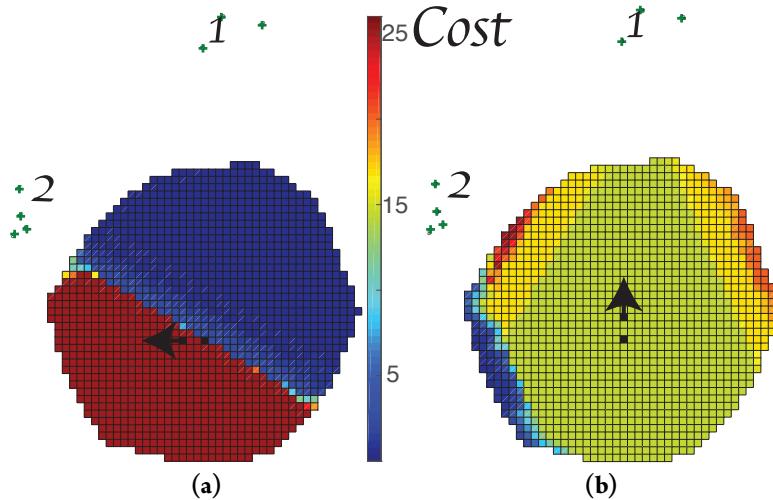


Figure 5.5.1: Spatial sensitivity to the ground truth location in respect to the objective value when considering "left" and "forward" actions accordingly. While X-BSP considers the weighted average of different possible measurements, denoted by colored area, ML-BSP considers only the most likely measurement, resulted from the black square.

a single planning session, for which expectation and ML produced different optimal actions.

Consider a robot with initial estimated location and covariance, given two candidate actions "left" and "forward". The world consists out of two types of landmarks, the first with high covariance and the second with low. Figures 5.5.1a-5.5.1b present the spatial cost values which are the result of choosing "left" or "forward" actions accordingly, and where warm colors denote higher cost values. Each pixel in Figures 5.5.1a-5.5.1b denotes a possible ground truth location of the robot, where the colored area represents the 1σ range of the prior covariance and the most likely state is denoted by a black square. While ML considers only the cost value resulting from the most likely state, expectation considers

multiple samples from different spatial locations. As a result expectation favored the "left" action while ML favored "forward". For $20k$ inference rollouts, each with a different ground truth location, choosing left is statistically favorable in the sense of minimizing cost (uncertainty), 74% of the times. So as expected we can deduce that X-BSP is statistically superior to ML-BSP.

Under continuous state space, the possibility of the robot location to match exactly the ML location is zero, and as can be seen from the cost values in Figure 5.5.1a a small shift in robot location could have drastic consequences over the cost value. Due to the fact that X-BSP takes different possible spatial locations into consideration, it provides a weighted estimate of the cost value that might be obtained, while ML-BSP consider a specific instance involving the ML location. After understanding the advantage X-BSP holds over ML-BSP one can ponder whether there might be some configuration allowing ML-BSP to match or at least get closer to the estimation performance of X-BSP. For example under the scenario presented above, adding more candidate actions to ML-BSP should improve the robustness of the estimated cost value. Of course this would result with a heavier computational load and some work is needed in order to determine how much of ML-BSP computational advantage is required to be sacrificed in favor of accuracy, we leave this for future work.

*If you can't describe what you are doing as a process,
then you don't know what you are doing.*

W. Edwards Deming

6

iX-BSP

IN THIS CHAPTER WE FORMULATE OUR INCREMENTAL EXPECTATION BSP (iX-BSP) APPROACH, WHICH ENABLES TO INCREMENTALLY CALCULATE THE OBJECTIVE FUNCTION BY RE-USING CALCULATIONS FROM PREVIOUS PLANNING SESSIONS, thus saving valuable computation time while at the same time preserving the benefits of the expectation solution provided by X-BSP. As explained in Section 5.3.2, the immediate rewards, required for calculating the objective value, are in the general case a function of candidate actions and future posterior beliefs calculated over sampled measurements. The way iX-BSP re-uses previous calculations is by enforcing specific measurements as opposed to sampling them from the appropriate measurement likelihood distribution. The measurements being enforced, were considered and sampled in some precursory planning session(s), in which each of the measurements had corresponding posterior belief and immediate reward. By enforcing some previously considered measurement, we can make use of the previously calculated posterior beliefs, instead of performing inference from scratch. In order to make use of the data acquired since these re-used beliefs have been calculated, when needed, we can incrementally update them to match the information up to current time.

This chapter is organized as follows. We first analyze the similarities between two successive planning sessions (Section 6.1), and use those insights as foundation to develop the paradigm for iX-BSP.

In Section 6.2 we provide an overview of the entire iX-BSP paradigm, and continue with covering each of the building blocks of iX-BSP: Selecting beliefs for re-use and deciding whether there is sufficient data for calculations re-use (Section 6.3), validating samples for re-use, incorporating forced samples and belief update (Section 6.4), calculating expectation incrementally with forced samples (Section 6.5). Section 6.6 provides a thorough analysis of iX-BSP, and a comparison to related work. Section 6.7 introduces a non-integral approximation to iX-BSP denoted as wildfire. Section 6.8 provides a thorough analysis of the affects wildfire holds over the iX-BSP solution. Section ?? demonstrates how iX-BSP can be utilized to improve existing approximations of X-BSP by introducing iML-BSP, denoting iX-BSP under the common ML assumption. Section ?? provides a thorough analysis of iML-BSP and a comparison to related work. Section 6.9 discusses some broader perspective of iX-BSP. Section 6.10 captivates the conclusions of this chapter along with possible extensions and usage. To improve coherence some theoretical background as well as proofs are covered in appendices.

It is worth mentioning that some of the following sections are accompanied by high-level algorithms, describing key aspects of iX-BSP. In an effort to simplify these algorithms for the readers' behalf, some of them are written in a sub-optimal manner (complexity-wise). When coming to implement iX-BSP, we trust the readers to adhere to the governing principles of iX-BSP while writing the source code in a complexity efficient manner.

6.1 COMPARING PLANNING SESSIONS

This section analyzes the similarities between two planning sessions that comply with Assumptions 1-3. In order to do so, let us consider two planning sessions, both with horizon of L steps ahead, the first occurred at time k and the second at time $k + l$. Under Assumption 2 both planning horizons overlap, i.e. $l < L$, and under Assumption 3 both planning sessions share some actions. For this comparison let us consider the action chosen at planning time k which also partially resides in a candidate action from planning time $k + l$, and denote both as $u_{k:k+L}^* = \{u_k^*, \dots, u_{k+L}^*\}$. Figure 6.1.1, illustrates the aforementioned horizon overlap between two beliefs at look ahead step at time t , given planning time k , i.e. $b[X_{t|k}]$, and given planning time $k + l$, i.e. $b[X_{t|k+l}]$, while the interesting shared sections, separated by time instances, are denoted as (i) (ii) and (iii).

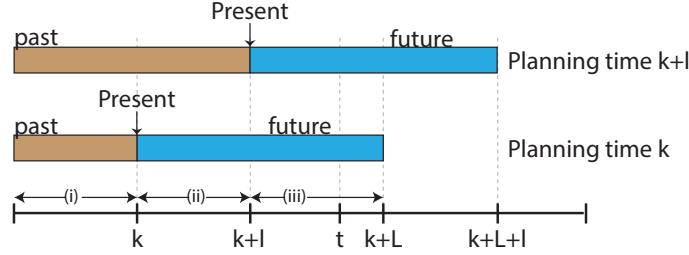


Figure 6.1.1: Horizon overlap between planning time k and planning time $k+L$, both with L steps horizon and same candidate actions:
(i) The shared history of both planning sessions (ii) The possibly outdated information of planning time k , since in planning time $k+L$ this time span is considered as known history (iii) Although in both it represents future prediction, it is conditioned over different history hence possibly different.

At future time $t \in [k+1, k+L]$, the belief created by the action sequence $\hat{u}_{k:t-1}^*$ is given by

$$b[X_{t|k}] \propto \underbrace{b[X_{k|k}]}_{(a)} \cdot \underbrace{\prod_{s=k+1}^{k+L} \left[\mathbb{P}(x_s | x_{s-1}, u_{s-1}^*) \prod_{g \in \mathcal{M}_{s|k}} \mathbb{P}(z_{s,g|k} | x_s, l_g) \right]}_{(b)} \cdot \underbrace{\prod_{i=k+L+1}^t \left[\mathbb{P}(x_i | x_{i-1}, u_{i-1}^*) \prod_{j \in \mathcal{M}_{i|k}} \mathbb{P}(z_{i,j|k} | x_i, l_j) \right]}_{(c)}, \quad (6.1)$$

where $(6.1)_{(a)}$ is the inference posterior at time k corresponding to the lower-bar area (i) in Figure 6.1.1, $(6.1)_{(b)}$ are the motion and observation factors of future times $k+1 : k+L$ corresponding to the lower-bar area (ii) and $(6.1)_{(c)}$ are the motion and observation factors of future times $k+L+1 : t$ corresponding to the lower-bar area (iii). For the same future time t and the same candidate action, the belief for planning time $k+L$ is given by,

$$b[X_{t|k+L}] \propto \underbrace{b[X_{k|k+L}]}_{(a)} \cdot \underbrace{\prod_{s=k+1}^{k+L} \left[\mathbb{P}(x_s | x_{s-1}, u_{s-1}^*) \prod_{g \in \mathcal{M}_{s|k+L}} \mathbb{P}(z_{s,g|k+L} | x_s, l_g) \right]}_{(b)} \cdot \underbrace{\prod_{i=k+L+1}^t \left[\mathbb{P}(x_i | x_{i-1}, u_{i-1}^*) \prod_{j \in \mathcal{M}_{i|k+L}} \mathbb{P}(z_{i,j|k+L} | x_i, l_j) \right]}_{(c)}, \quad (6.2)$$

where $(6.2)_{(a)}$ is the inference posterior at time k corresponding to the upper-bar area (i) in Figure 6.1.1, $(6.2)_{(b)}$ are the motion and observation factors of past times $k+1 : k+L$ corresponding to the upper-bar area (ii) and $(6.2)_{(c)}$ are the motion and observation factors of future times $k+L+1 : t$ corresponding to the upper-bar area (iii).

Although seemingly conditioned on a different history (k vs $k+L$), $(6.1)_{(a)}$ and $(6.2)_{(a)}$ are identi-

cal and denote the same posterior obtained at time k (see Figure 6.1.1 area (i)), leaving the difference between (6.1) and (6.2) restricted to $(\cdot)_{(b)}$ and $(\cdot)_{(c)}$. While $(6.1)_{(b)}$ represents future actions and future measurements predicted at time k , $(6.2)_{(b)}$ represents executed actions and previously acquired measurements, this can be seen more clearly using area (ii) in Figure 6.1.1. At planning time k (i.e. lower bar), area (ii) denotes future prediction for the time interval $k : k + l$, while at planning time $k + l$ (i.e. upper bar), the same time interval denotes past measurements, and so $(6.1)_{(b)}$ and $(6.2)_{(b)}$ are potentially different, depending on how accurate was the prediction at planning time k . As thoroughly explained in Section 4.3.3, the difference between the predictions made during planning and the actual measurements obtained in present time is twofold, the difference in measurement values (see Section 4.4) and the difference in data association (see Section 4.6).

Even-though both $(6.1)_{(c)}$ and $(6.2)_{(c)}$ refer to future actions and measurements (see area (iii) in Figure 6.1.1) they do so with possibly different values and data association since they were sampled from possibly different probability densities. Solving the objective (5.11), requires sampling from (5.12) (e.g. using Alg. 2), the samples from planning time k were sampled from $\mathbb{P}(z_{t|k} | H_{t|k}^-)$, while the samples from planning time $k + l$ were sampled from $\mathbb{P}(z_{t|k+l} | H_{t|k+l}^-)$. These probabilities would be identical only if conditioned on the same history, i.e. only if the predictions made for time interval $k : k + l$ at planning time k were accurate both in data association and measurement values.

As such, in order to mind the gap between (6.1) and (6.2), and obtain identical expressions one must update $(6.1)_{(b)}$ to match $(6.2)_{(b)}$, and second, to adjust the samples from $(6.1)_{(c)}$ to properly represent the updated measurement probability density.

6.2 iX-BSP OVERVIEW

This section presents an overview of iX-BSP at planning time $k + l$, whilst the relevant precursory planning session occurred at planning time k , as summarized in Alg. 3. For the reader's convenience all the notations of this section are summarized in Table 6.1. After executing l steps out of the (sub)optimal action sequence suggested by planning at time k , and performing inference over the newly received measurements, we obtain $b[X_{k+l|k+l}]$. Performing planning at time $k + l$ under iX-BSP, requires first deciding on the planning sub-tree from the precursory planning session to be considered for re-use (Alg. 3 line 1). Considering belief roots of candidate planning sub-trees, the selected sub-tree is the one with the "closest" belief root to $b[X_{k+l|k+l}]$, i.e. the one with the minimal distance to it while considering some appropriate probability density function distance. We denote the closest belief root and the appropriate planning sub-tree as $\tilde{b}[X_{k+l|k}]$ and $\mathcal{B}_{k+l|k}$ respectively.

In case the distance of the closest belief (denoted by Dist) is larger than some critical value ε_c , i.e. the closest prediction from the precursory planning session is too far off, iX-BSP would presumably

have no advantage over the standard X-BSP so the latter is executed (Alg. 3 line 12). On the other hand, if Dist is smaller than the critical value ε_{wf} , we consider the difference between the beliefs as insignificant and continue with re-using the precursory planning session without any additional update (Alg. 3 line 4). We denote the aforementioned as wildfire. When the precursory planning is close-enough (see Figure 6.3.2), we can appropriately re-use it to save valuable computation time. While we go further and elaborate on specific methods we used in this work, e.g. determining belief distance (see Section 6.3) or representative sample (see Section 6.4.2), iX-BSP is indifferent to any specific method, as long as it serves its intended purpose.

Algorithm 3 iX-BSP: Planning time $k+1$

Input:

data	▷ Calculations used for the precursory planning session
$b[X_{k+l k+l}]$	▷ The up-to-date inference posterior for time $k+l$
useWF, $\varepsilon_c, \varepsilon_{wf}$	▷ User defined flags & thresholds

```

1: Dist,  $\mathcal{B}_{k+l|k} \leftarrow \text{SELECTCLOSESTBRANCH}(b[X_{k+l|k+l}], \text{data})$            ▷ see Section 6.3.1
2: if Dist  $\leq \varepsilon_c$  then                                                 ▷ belief distance threshold  $\varepsilon_c$ 
3:   if useWF  $\cap$  (Dist  $\leq \varepsilon_{wf}$ ) then                                     ▷ wildfire threshold  $\varepsilon_{wf}$ 
4:     data  $\leftarrow \mathcal{B}_{k+l|k}$  ▷ Reusing the entire selected branch without any update, see Section 6.7
5:   else
6:     data  $\leftarrow \text{INCUPDATEBELIEFTREE}(\mathcal{B}_{k+l|k})$                                 ▷ see Section 6.4
7:   end if
8:   data  $\leftarrow$  perform X-BSP over horizon steps  $k+L+1 : k+L+1$ 
9:   Solve Eq. 5.11, for each candidate action                                     ▷ see Section 6.5
10:   $u_{k+i:k+L|k+i}^* \leftarrow$  find best action
11: else
12:    $u_{k+i:k+L|k+i}^* \leftarrow$  perform X-BSP( $b[X_{k+l|k+l}]$ )
13: end if
14: return  $u_{k+i:k+L|k+i}^*$ , data

```

The planning sub-tree $\mathcal{B}_{k+l|k}$ is comprised of all future beliefs, i.e. $k+1 : k+L$, calculated as part of the planning session from time k , which originate in $\tilde{b}[X_{k+l|k}]$. We update these beliefs with the information received in inference between time instances $k+1$ and $k+l$, and selectively re-sample predicted measurements (line 6) in an effort to maintain a representative set of samples for the nominal distribution. In case one of the aforementioned beliefs also meets the wildfire condition (see Section 6.7) we consider it, and all of its descendants as already updated. Once the update is complete, we have a planning horizon of just $L-l$ steps, i.e. to the extent of the horizon overlap, hence we need to calculate the rest from scratch, i.e. perform X-BSP for the final l steps (line 8).

We are now in position to update the immediate reward function values and calculate their expected value in search of the (sub)optimal action sequence (line 10), thus completing the planning for time $k + l$.

Since we are re-using samples from different planning sessions at planning time $k + l$ we are required to compensate for the different measurement likelihood, through proper formulation. In the sequel we show that our problem falls within the Multiple Importance Sampling problem (see Appendix D), so we estimate the expected reward values using importance sampling based estimator, thus completing the planning for time $k + l$.

Differently from X-BSP which returns only the selected action sequence, iX-BSP is also required to return more data from the planning process in order to facilitate re-use (line 14).

Variable	Description
$\square_{t k}$	Of time t while current time is k
$\mathcal{M}_{t k}$	Data Association at time t while current time is k
$b[X_{t k}]$	belief at time t while current time is k
$b^{-}[X_{t k}]$	belief at time $t - 1$ propagated only with action $u_{t-1 k}$
$\mathcal{B}_{k k}$	The entire belief tree from planning at time k
$\tilde{b}[X_{t k}]$	The root of the selected branch for re-use in planning at time t
$\mathcal{B}_{t k}$	The set of all beliefs from planning time k rooted in $\tilde{b}[X_{t k}]$
Dist	The distance between $\tilde{b}[X_{t k}]$ and the corresponding posterior $b[X_{t t}]$
data	All available calculations from current and precursory planning session
$u_{k:k+L k}^*$	The (sub)optimal action sequence of length L chosen in planning at time k
ε_c	belief distance critical threshold, above it re-use has no computational advantage
ε_{wf}	wildfire threshold, bellow it distance is considered close-enough for re-use without any update
useWF	a binary flag determining whether or not the wildfire condition is considered
$\mathbb{D}(.)$	belief divergence / metric

Table 6.1: Notations for Sections 6.2-6.3

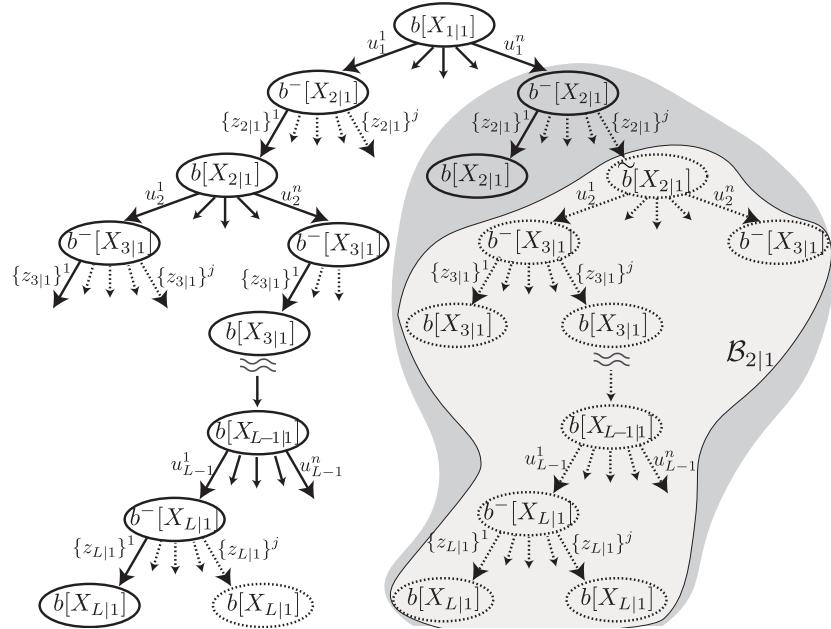


Figure 6.3.1: X-BSP performs lookahead search on a tree with depth L . Each belief tree node represents a belief. For each node, the tree branches either for a candidate action or a sampled measurement. The corresponding belief tree for ML-BSP is marked with solid lines, while the dashed lines represent the parts of X-BSP that relate to sampled measurements. Under iX-BSP, the gray-marked parts of the tree are being re-used for the succeeding planning session.

6.3 SELECTING BELIEFS FOR RE-USE

This section covers an integral part of iX-BSP, dealing with how to select which beliefs to re-use, and from where. At each step along the planning horizon, iX-BSP is required to choose beliefs for re-use. Our goal is to minimize any required updates, i.e. the beliefs we would like to re-use should be as "close" as possible to the beliefs we would have obtained through standard X-BSP. In iX-BSP, as well as in X-BSP, the number of beliefs per future time step is derived from the number of samples per action per time step; in order to re-use previous calculations while avoiding a computational load, we need to choose which beliefs to consider for re-use. The need to obtain the closest belief for re-use entails three fairly complicated problems: Where to find it (Section 6.3.1), how to find it (Section 6.3.2), and how to determine what is considered "close" in belief space (Section 6.3.3). For the reader's convenience all the notations of this section are summarized in Table 6.1.

6.3.1 SELECTING THE CANDIDATE SET FOR RE-USE

While every set of previously calculated beliefs can serve as potential candidates for re-use, in this work we consider previous planning sessions as they are readily available. It is worth mentioning that the problem of searching a set of candidate beliefs can be computationally expensive, thus poten-

tially sabotaging the efforts of iX-BSP to relieve the computational load of BSP. In order to avert from directly dealing with the aforementioned "search problem" within belief space and maximize the chances of finding an adequate candidate for re-use, we introduce Assumptions 2-3 that are not an integral part of iX-BSP. Following Assumption 2 we assure that the previous planning session has some overlapping horizon with the current planning session, hence increasing the chances of locating a "close enough" belief for re-use.

Using Assumption 3 we can prune the full belief-tree from previous planning time k , denoted as $\mathcal{B}_{k|k}$, and consider only a subset of it while assuring overlapping of some candidate actions. More specifically, we prune $\mathcal{B}_{k|k}$ to consider the sub-tree $\mathcal{B}_{k+l|k} \subset \mathcal{B}_{k|k}$, which is rooted in $\tilde{b}[X_{k+l|k}]$ such that

$$\tilde{b}[X_{k+l|k}] = \arg \min_{b[X_{k+l|k}] \in \mathcal{B}_{k|k}} \mathbb{D}(b[X_{k+l|k}], b[X_{k+l|k+l}]), \quad (6.3)$$

where $\mathbb{D}(\cdot)$ is a metric (or divergence) quantifying the difference between two beliefs (see Section 6.3.3), $b[X_{k+l|k+l}]$ is the posterior from inference at time $k + l$, and $\tilde{b}[X_{k+l|k}]$ is one of the beliefs for lookahead step l of planning at time k . This search for the closest belief is performed by Alg. 4 and discussed in Section 6.3.2. By considering the sub-belief tree of the closest prediction from planning at time k to the current posterior, we ensure minimal required update along the lookahead steps, i.e. minimizing the difference between the prediction (6.1)_(b) and what eventually happened (6.2)_(b).

Without loss of generality, we now make use of Figure 6.3.1 to illustrate the branch selection process, i.e. how we choose a candidate set of beliefs given an entire previous planning tree. Figure 6.3.1 illustrates a belief tree of X-BSP at planning time $t = 1$ for a horizon of L steps, with n candidate actions and j sampled measurements per step, resulting with $(n \cdot j)^1$ different beliefs for future time $t = 2$ and $(n \cdot j)^{L-1}$ for future time L . Let us assume action u_1^n has been determined as optimal at planning time $t = 1$ and has been executed. After attaining new measurements for current time $t = 2$ and calculating the posterior belief $b[X_{2|2}]$, we perform planning once more. Now, under iX-BSP, instead of calculating everything from scratch we would like to re-use previous calculations; specifically, under Assumption 2 we consider the beliefs calculated at planning time $t = 1$. Instead of considering the entire tree $\mathcal{B}_{1|1}$ for re-use, we look for some sub-tree $\mathcal{B}_{2|1} \subset \mathcal{B}_{1|1}$ rooted in $\tilde{b}[X_{2|1}]$ such that

$$\tilde{b}[X_{2|1}] = \arg \min_{b[X_{2|1}] \in \mathcal{B}_{1|1}} \mathbb{D}(b[X_{2|1}], b[X_{2|2}]). \quad (6.4)$$

We start by considering all beliefs $\{b^i[X_{2|1}]\}_{i=1}^{n \cdot j}$ meeting Assumption 3, i.e. all beliefs marked by the dark gray area in Figure 6.3.1 which considered the same action sequence. Now, from the remaining j beliefs, using Alg. 4 (see Section 6.3.2) we denote the closest belief to the posterior $b[X_{2|2}]$ as $\tilde{b}[X_{2|1}]$.

Once we determined $\tilde{b}[X_{2|1}]$, we define the closest branch as consisting of all the beliefs rooted in $\tilde{b}[X_{2|1}]$, and denote it as $\mathcal{B}_{2|1}$, marked in Figure 6.3.1 by the light gray area. In case there are no beliefs in the set $\{b'[X_{2|1}]\}_{i=1}^{n-j}$ meeting Assumption 3, we will need to search through the entire set for the closest belief $\tilde{b}[X_{2|1}]$. In the following section we describe how the closest belief is located.

6.3.2 FINDING THE CLOSEST BELIEF

This section covers the problem of how to locate the closest belief given a set of candidate beliefs, as required when selecting the closest branch for re-use (Section 6.3.1, Alg. 3 line 1) or when incrementally updating the belief tree under iX-BSP (Alg. 5 line 10). As part of our problem, we have a set of candidate beliefs for re-use, denoted as $\mathcal{B}_{k+l|k}$, and some posterior $b[X_{i|k+l}]$ we wish to be close to. Our goal is to find within $\mathcal{B}_{k+l|k}$ the closest belief to $b[X_{i|k+l}]$, where $i > k + l$ denote some lookahead step. Locating the closest belief requires quantifying the differences between two beliefs into a scalar distance. We denote the distance function, whether a metric or a divergence, by $\mathbb{D}(.)$. Let us consider some candidate belief $b[X_{i|k}] \in \mathcal{B}_{k+l|k}$, although referring to the same future time i as $b[X_{i|k+l}]$, it is conditioned on different history, and therefore is potentially different. While Section 6.1 discussed the reasons for such difference between $b[X_{i|k}]$ and $b[X_{i|k+l}]$, here we quantify this difference using a belief distance. Projecting $b[X_{i|k}]$ into our belief distance space yields a point that suggests how different is $b[X_{i|k}]$ from $b[X_{i|k+l}]$. After projecting all candidate beliefs from $\mathcal{B}_{k+l|k}$ into the belief distance space in reference to $b[X_{i|k+l}]$, the problem of locating the closest belief to $b[X_{i|k+l}]$ is reduced to a problem of locating the nearest neighbor.

Algorithm 4 ClosestBelief

Input:

$\mathcal{B}_{k+l k}$	▷ set of candidate beliefs for re-use from planning at time k , see Section 6.3.1
$b[X_{i+1 k+l}]$	▷ The belief to check distance to, from planning at time $k + l$

```

1:  $\delta_{min} = \infty$ 
2: for  $b[X_{i+1|k}] \in \mathcal{B}_{k+l|k}$  do
3:    $\delta \leftarrow \mathbb{D}(b[X_{i+1|k}], b[X_{i+1|k+l}])$  ▷ probability metric/ divergence to determine belief distance
4:   if  $\delta \leq \delta_{min}$  then                                ▷ keeping track over the shortest distance
5:      $\delta_{min} \leftarrow \delta$ 
6:      $b'[X_{i+1|k}] \leftarrow b[X_{i+1|k}]$ 
7:   end if
8: end for
9: return  $\delta_{min}, b'[X_{i+1|k}]$ 

```

While there are more efficient ways to determine the closest belief, we made use of a simple realization of $\text{BeliefDist}(\cdot)$ in Alg. 4. Any other, more efficient realization, that provides with the same end-result is acceptable, and would benefit the computational load reduction of iX-BSP. Alg. 4 determines the closest belief given a set of beliefs $\mathcal{B}_{k+l|k}$ and a target belief $b[X_{i|k+l}]$, by simply calculating the distance between the target belief to each belief in the set $\mathcal{B}_{k+l|k}$ using $\mathbb{D}(\cdot)$, and picking the closest one. The minimal distance associated with the closest belief is thus given by

$$\delta_{\min} = \min_{b[X_{i|k}] \in \mathcal{B}_{k+l|k}} \mathbb{D}(b[X_{i|k}], b[X_{i|k+l}]), \quad (6.5)$$

while the distinction whether δ_{\min} is acceptable or not, happens outside of Alg. 4 (see Alg. 3 lines 2-3 and Alg. 5 lines 11-12) as discussed next.

6.3.3 WHAT IS CLOSE ENOUGH

Choosing a candidate belief with minimal distance is not enough, as this candidate belief could still be very different, and thus require a substantial computational effort in order to update. In order to deal with this issue, we need to set some criteria over the belief distance. Let us consider a belief metric space, in-which each point is a unique projection of a candidate belief $b[X_{i|k+l}]$ denoting the distance between the aforementioned candidate belief and $b[X_{i|k+l}]$. Figure 6.3.2 illustrates such space, where

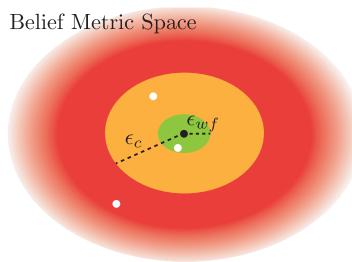


Figure 6.3.2: Illustration of the relative belief distance space. Each point in this space represents some belief $b[X_{i|k}]$, where the black dot denotes $b[X_{i|k+l}]$ as the origin. All beliefs $b[X_{i|k}]$ close to the origin up to $\epsilon_{w,f}$, i.e. in the green zone, are being re-used without any update. All beliefs $b[X_{i|k}]$ close to the origin up to ϵ_c but farther than $\epsilon_{w,f}$, i.e. in the orange zone, are being re-used with some update. All beliefs $b[X_{i|k}]$ that are more than ϵ_c away from the origin, i.e. in the red zone, are considered as not close enough to make a re-use worth while.

the black dot represents the homogeneous projection $\mathbb{D}(b[X_{i|k+l}], b[X_{i|k+l}])$, and the rest of the points denote $\mathbb{D}(b[X_{i|k}], b[X_{i|k+l}])$ e.g. the three white dots in Figure 6.3.2. We divide distances around the

homogeneous projection into three areas,

$$\begin{cases} \mathbb{D}(b[X_{i|k}], b[X_{i|k+l}]) \leq \varepsilon_{wf} & \text{close enough for re-use "as is", see Section 6.7} \\ \varepsilon_{wf} < \mathbb{D}(b[X_{i|k}], b[X_{i|k+l}]) \leq \varepsilon_c & \text{close enough for re-use} \\ \varepsilon_c < \mathbb{D}(b[X_{i|k}], b[X_{i|k+l}]) & \text{too far off for re-use} \end{cases} \quad (6.6)$$

denoted respectively in green orange and red. When wildfire is not enabled, i.e. `useWF` = false, the belief metric space is divided into two areas (orange and red), separated by a single parameter ε_c . The case where wildfire is enabled, i.e. `useWF` = true, is covered in Section 6.7. At this point we leave the procedure of choosing ε_c and ε_{wf} for future work, and consider it as a heuristic. In the following we do however show analytically (see Section 6.7.3) and empirically (see Section 6.8.2) the connection between ε_{wf} and the objective value.

Each belief metric (divergence) would result with a possibly different projection onto metric space, hence with probably different values for ε_c and ε_{wf} . As part of our work we considered several alternatives for belief distance, a DA based divergence and another based on Jeffreys divergence.

THE \mathbb{D}_{DA} DISTANCE

Under \mathbb{D}_{DA} we start by sorting all candidate beliefs according to data association (DA) differences, looking for the smallest available difference. For example, the DA differences between Eq. (6.1) and Eq. (6.2) are given by matching their DA data denoted by \mathcal{M} . Such matching would yield three possible differences: the DA that has been correctly predicted and need not be changed

$$\mathcal{M}_{k+1:k+l|k} \cap \mathcal{M}_{k+1:k+l|k+l}, \quad (6.7)$$

the DA that has been wrongfully predicted and need to be removed

$$\mathcal{M}_{k+1:k+l|k} \setminus \mathcal{M}_{k+1:k+l|k+l}, \quad (6.8)$$

and the DA that has not been predicted and need to be added

$$\mathcal{M}_{k+1:k+l|k+l} \setminus \mathcal{M}_{k+1:k+l|k}. \quad (6.9)$$

In case there is more than a single belief with minimal DA difference, we continue to sort the remaining beliefs according to the difference between values of corresponding predicted measurements and similarly look for the minimal difference. In case there is more than a single belief with minimal mea-

surement value difference, we select arbitrarily out of the remaining beliefs, and consider the chosen belief as the closest one. A detailed explanation of the DA matching process can be found in Section 4.6.2. It is worth stressing that \mathbb{D}_{DA} is just a divergence and not a metric, as it does not meet the symmetry and sub-additivity requirements.

THE $\mathbb{D}_{\sqrt{J}}$ DISTANCE

The $\mathbb{D}_{\sqrt{J}}$ distance is a variant of the Jeffreys divergence presented by [31] (see Appendix E). This is a symmetric divergence for general probabilities that also has a special form in the case of Gaussian beliefs (for full derivation see Appendix E). For two Gaussian beliefs $b[X_{t|k+l}] \sim \mathcal{N}(\mu_p, \Sigma_p)$ and $b[X_{t|k}] \sim \mathcal{N}(\mu_q, \Sigma_q)$, the $\mathbb{D}_{\sqrt{J}}$ distance between them is given by,

$$\begin{aligned} \mathbb{D}_{\sqrt{J}}(b[X_{t|k+l}], b[X_{t|k}]) = \\ \frac{1}{4} \sqrt{(\mu_p - \mu_q)^T [\Sigma_q^{-1} + \Sigma_p^{-1}] (\mu_p - \mu_q) + \text{tr}(\Sigma_q^{-1} \Sigma_p) + \text{tr}(\Sigma_p^{-1} \Sigma_q) - d_p - d_q}, \end{aligned} \quad (6.10)$$

where d_p and d_q are the joint state dimension of $b[X_{t|k+l}]$ and $b[X_{t|k}]$, respectively.

6.4 INCREMENTAL UPDATE OF BELIEF-TREE

In Section 6.3.1 we determined the candidate set $\mathcal{B}_{k+l|k}$, and using Section 6.3.2 have the ability to locate for each posterior belief $b[X_{i|k+l}]$ the closest belief in $\mathcal{B}_{k+l|k}$. Now, we can focus on one of our main contributions, incrementally creating a belief tree, through the re-use of previously calculated beliefs, while accounting for all information differences. To this end we supply Alg. 5, tasked with creating the belief tree of planning time $k+l$ through selective re-use of beliefs from $\mathcal{B}_{k+l|k}$. The process starts with the posterior $b[X_{k+l|k+l}]$, and continues with every new belief $b^s[X_{i|k+l}]$ that is added to the new belief tree up to future time $k+L$, where s accommodates all different sampled beliefs at future time i . For the reader's convenience all the notations of this section are summarized in Table 6.1.

We will now describe a single iteration of Alg. 5, in which we have reached the s_{th} belief (line 2) at the i_{th} lookahead step (line 1), and already handled all previous steps and beliefs. First, we check whether the belief $b^s[X_{i|k+l}]$ has been created under the wildfire condition (line 3), i.e. directly taken from $\mathcal{B}_{k+l|k}$ without any update (see Section 6.7). In case it did, we continue to take its descendants directly from the appropriate beliefs in $\mathcal{B}_{k+l|k}$ without any update (line 5) and mark them as created under the wildfire condition (line 6). For the case where the belief $b^s[X_{i|k+l}]$ has not been created

Algorithm 5 IncUpdateBeliefTree

Input:

$$\begin{aligned} \mathcal{B}_{k+l|k} \\ b[X_{k+l|k+l}] \end{aligned}$$

▷ The selected branch, see Section 6.3.1
▷ The posterior from precursory inference

```

1: for each  $i \in [k+l, k+L-1]$  do                                ▷ each overlapping horizon step
2:   for each  $s \in [1, n_u(n_x \cdot n_z)^{i-k-l}]$  do          ▷ each belief in the  $i_{th}$  horizon step
3:     if useWF  $\cap$  ISWILDFIRE( $b^s[X_{i|k+l}]$ ) then
4:        $r_o \leftarrow (s-1) \cdot n_u \cdot (n_x \cdot n_z)$ 
5:        $\{b^r[X_{i+1|k+l}]\}_{r=r_o+1}^{r_o+n_u \cdot (n_x \cdot n_z)} \leftarrow$  all first order children of  $b^{s'}[X_{i|k}]$       ▷ see Section 6.7
6:       mark all  $\{b^r[X_{i+1|k+l}]\}_{r=r_o+1}^{r_o+n_u \cdot (n_x \cdot n_z)}$  as wildfire
7:     else
8:       for each candidate action  $a \in [1, n_u]$  do
9:          $b_a^{s-}[X_{i+1|k+l}] \leftarrow$  propagate  $b^s[X_{i|k+l}]$  with candidate action  $a$ 
10:        dist,  $b_a^{s'-}[X_{i+1|k}] \leftarrow$  CLOSESTBELIEF( $\mathcal{B}_{k+l|k}$ ,  $b_a^{s-}[X_{i+1|k+l}]$ ) ▷ see Section 6.3.2
11:        if dist  $\leq \varepsilon_c$  then                                ▷ re-use condition
12:          if useWF  $\cap$  (dist  $\leq \varepsilon_{wf}$ ) then          ▷ wildfire condition
13:             $\{b_a^r[X_{i+1|k+l}]\}_{r=1}^{n_x \cdot n_z} \leftarrow$  all first order children of  $b_a^{s'-}[X_{i+1|k}]$           ▷
                                         see Section 6.7
14:            mark  $\{b_a^r[X_{i+1|k+l}]\}_{r=1}^{n_x \cdot n_z}$  as wildfire
15:            Continue with next candidate action (i.e. jump to line 8)
16:        else
17:          samples  $\leftarrow$  all samples taken from  $b_a^{s'-}[X_{i+1|k}]$ 
18:          {repSamples} $_1^{n_x \cdot n_z}$ , data  $\leftarrow$  IsREPSAMPLE(samples,  $b_a^{s-}[X_{i+1|k+l}]$ ) ▷
                                         see Section 6.4.2
19:        end if
20:      else                                     ▷ not computationally effective to re-use, resample all
21:        {repSamples} $_1^{n_x \cdot n_z}$ , data  $\leftarrow (n_x \cdot n_z)$  fresh samples based on  $b_a^{s-}[X_{i+1|k+l}]$  ▷
                                         see Alg. 2
22:      end if
23:      data  $\leftarrow$  UPDATEBELIEF(dist, {repSamples} $_1^{n_x \cdot n_z}$ , data) ▷ see Section 6.4.3
24:      data  $\leftarrow$  update reward(cost) values for action  $a$           ▷ see Section 6.4.4
25:    end for
26:  end if
27: end for
28: end for
29: return data

```

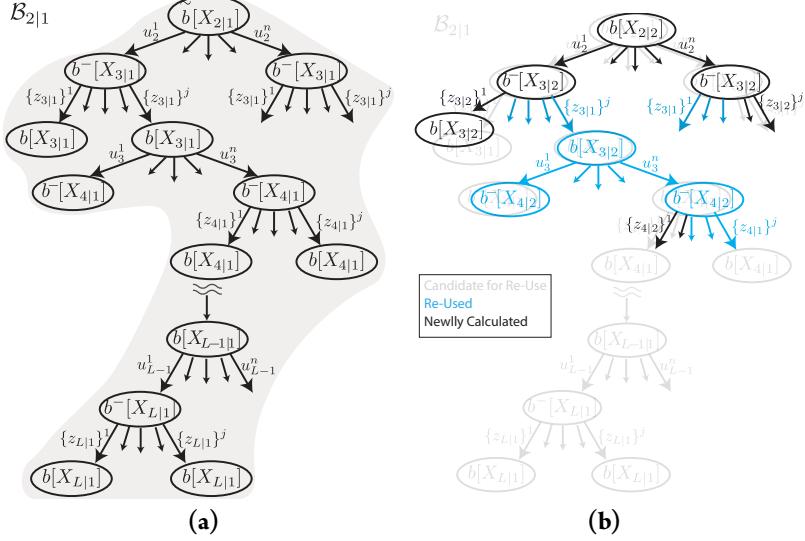


Figure 6.4.1: The belief update process of iX-BSP presented in a belief tree, where each node represents a belief that branches either for one of n candidate actions or j sampled measurements. The selected branch for re-use from Figure 6.3.1, denoted by $\mathcal{B}_{2|1}$, is presented in (a) and as a water mark in (b). The succeeding iX-BSP planning session at time $t = 2$ is illustrated in (b), where the re-used sampled measurements and succeeding beliefs are marked in light blue.

under the wildfire condition we propagate it with the α candidate action, where $\alpha \in [1, n_u]$ and n_u denotes the number of candidate actions per time step (line 9); this propagation yeilds $b_a^{s^-}[X_{i+1|k+l}]$. We then consider all propagated beliefs $b^-[X_{i+1|k}] \subset \mathcal{B}_{k+l|k}$, and search for the closest one to $b_a^{s^-}[X_{i+1|k+l}]$ in the sense of belief distance, as discussed in Section 6.3.2. Once found, we denote the closest propagated belief as $b_a^{s'}^-[X_{i+1|k}]$ (line 10).

In case there is no such belief close enough to make the update worthwhile, i.e. $\text{dist} > \varepsilon_c$, for this candidate action we continue as if using X-BSP (line 21). In case the distance of the closest belief meets the wildfire condition ε_{wf} , we consider all beliefs $b[X_{i+1|k}] \subset \mathcal{B}_{k+1|k}$ that are rooted in $b^{*-}[X_{i+1|k}]$. Otherwise we continue and check whether the samples generated using $b_a^{*-}[X_{i+1|k}]$ constitute an adequate representation for $\mathbb{P}(z_{i+1|k+l}|H_{i|k+l}, u_{i|k+l}^a)$, and re-sample if needed (line 18, see Section 6.4.2). Once we obtain the updated set of samples $\{\text{repSamples}\}_1^{n_x n_z}$, whether all were freshly sampled, entirely re-used or somewhere in between, we can acquire the set of posterior beliefs for look ahead step $i + 1$ $\{b_a[X_{i+1|k+l}]\}_1^{n_x n_z}$ (line 23) through an update, as discussed subsequently (see Section 6.4.3). Once we have all updated beliefs for future time $i + 1$, we can update the reward values of each of which (see Section 6.4.4). We repeat the entire process for the newly acquired beliefs $\{b_a[X_{i+1|k+l}]\}_1^{n_x n_z}$, and so forth up to $k + L$.

Without loss of generality in the supplied formulation of Alg. 5, the candidate beliefs are considered only to the extent of the planning horizon overlap, i.e. $k + l + 1 : k + L$, whereas beliefs for the rest of the horizon $k + L + 1 : k + L + l$, are obtained by performing X-BSP (Alg. 3, line 8).

Variable	Description
$\square_{t k}$	Of time t while current time is k
$b[X_{t k}]$	belief at time t while current time is k
$b^-[X_{t k}]$	belief at time $t - 1$ propagated only with action $u_{t-1 k}$
$\tilde{b}[X_{t k}]$	The root of the selected branch for re-use from planning time k
$\mathcal{B}_{t k}$	The set of all beliefs from planning time k rooted in $\tilde{b}[X_{t k}]$
$b^s[X_{t k+l}]$	The s_{th} sampled belief representing $b[X_{t k+l}]$
$b_a^s[X_{t+1 k+l}]$	The sampled belief $b^s[X_{t k+l}]$ propagated with the a candidate action
$\{b_a^r[X_{t k+l}]\}_{r=1}^n$	A set of n sampled beliefs that are first order children of $b_a^s[X_{t k+l}]$ and are representing $b[X_{t k+l}]$
$b_a^{s'}[X_{t+i k}]$	A propagated belief from $\mathcal{B}_{t k}$ closest to $b_a^s[X_{t+i k+l}]$
dist	The distance between $b_a^{s'}[X_{t k}]$ and $b_a^s[X_{t k+l}]$
$\{b^r[X_{t k}]\}_{r=1}^n$	A set of n sampled beliefs representing $b[X_{t k}]$
n_u	number of candidate actions per step
$(n_x \cdot n_z)$	number of samples for each candidate action
data	All available calculations from current and precursory planning session
ε_c	belief distance critical threshold, max distance for re-use computational advantage
ε_{wf}	wildfire threshold, max distance to be considered as close-enough for re-use without any update
useWF	a binary flag determining whether or not the wildfire condition is considered
β_σ	σ acceptance range parameter, for considering measurements as representative
$\mathbb{D}_{\sqrt{J}}(p, q)$	The distance between distributions p and q according to the $\mathbb{D}_{\sqrt{J}}$ distance
$D_{DA}(p, q)$	The divergence between distributions p and q according to the data association difference

Table 6.1: Notations for Section 6.4

Next, we provide a walk-through example for Alg. 5 (Section 6.4.1) elaborate on belief distance (Section 6.3.2), and continue with covering key aspects required by Alg. 5: determining whether samples are representative or not (Section 6.4.2), the process of belief update given the representative set of samples $\{\text{repSamples}\}_1^n$ (Section 6.4.3) and the incremental calculation of the immediate reward values per sampled belief (Section 6.4.4).

6.4.1 ALG. 5 WALK-THROUGH EXAMPLE

We will now demonstrate Alg. 5 using Figure 6.4.1. Figure 6.4.1a illustrates the selected branch for re-use from a precursory planning session at time $t = 1$, i.e. $\mathcal{B}_{2|1}$ (see Figure 6.3.1), we have n candidate actions each step, and for each candidate action we have j sampled measurements. Figure 6.4.1b illustrates a part of the belief tree at time $t = 2$, where the top of the tree is the posterior belief of current time $t = 2$, i.e. $b[X_{2|2}]$, from which we start the algorithm. Since $b[X_{2|2}]$ is the posterior of current time $t = 2$, it was not created under the wildfire condition, so we jump directly to line 8. We propagate $b[X_{2|2}]$ with each of the n candidate actions starting with u_2^1 , and obtain the left most belief in the second level of the tree, $b^-[X_{3|2}]$ (line 9). Using BeliefDist(.) we obtain the closest belief from $\mathcal{B}_{2|1}$ to $b^-[X_{3|2}]$ as well as their distance dist. For our example the closest belief turns out to be the one which has been also propagated by the same candidate action, i.e. the left most $b^-[X_{3|1}]$ in the second level of $\mathcal{B}_{2|1}$. Since the distance suggests re-use is worthwhile but does not meet the wildfire condition, i.e. $\varepsilon_{wf} < \text{dist} \leq \varepsilon_c$, we proceed to line 17 in Alg. 5.

We denote the set (of sets) of all j sampled measurements as samples, i.e.

$$\text{samples} \leftarrow \{\{z_{3|1}\}^1, \dots, \{z_{3|1}\}^j\}.$$

Using IsRepSample(.) we obtain a representative set for the measurement likelihood $\mathbb{P}(z_{3|2}|H_{2|2}, u_2^1)$ (see Section 6.4.2). As we can see in Figure 6.4.1b, other than $\{z_{3|1}\}^1$, which has been re-sampled, all other samples are re-used (denoted by blue arrows). Once we have a representative set of measurements, which in our case all but one are re-used from planning time $t = 1$, we can update the appropriate beliefs using UpdateBelief(.) (see Section 6.4.3). The belief resulting from the newly sampled measurement $\{z_{3|2}\}^1$ is calculated from scratch by adding the measurement to $b^-[X_{3|2}]$ and performing inference, while the re-used samples allow us to incrementally update the appropriate beliefs from $\mathcal{B}_{2|1}$, rather than calculate them from scratch (see Section 6.4.3).

We now have an updated set of beliefs for future time $t = 3$, that considers the candidate action u_2^1 . For each of the aforementioned beliefs we incrementally calculate the appropriate reward value (see Section 6.4.4) thus completing the incremental update for candidate action u_2^1 . We repeat the aforementioned for the rest of the candidate actions, thus completing the third level of the belief tree presented in Figure 6.4.1b. In a similar manner we continue to incrementally calculate the deeper levels of the belief tree up to future time $t = L$, thus concluding Alg. 5.

6.4.2 REPRESENTATIVE SAMPLE

This section covers the problem of obtaining a set of measurement samples that are representative of the measurement likelihood distribution we should have sampled from. The motivation for re-using previously sampled measurements lies within the desire to refrain from performing inference by re-using previously calculated beliefs. As explained in Section 6.1, assuming the differences between (6.1)_(b) and (6.2)_(b) have been resolved (the predicted factors and their counterparts that already have been obtained respectively), the difference between Eq. (6.1) to Eq. (6.2) is limited to the difference between (6.1)_(c) and (6.2)_(c). Assuming both use the same action sequence, the difference between (6.1)_(c) and (6.2)_(c) is limited to the predicted measurements being considered by each. While the field of representative sampling is a rich research area on its own, in order to facilitate iX-BSP, we chose a straightforward approach that can be easily substituted with a more sophisticated one in due time. Under the sampling paradigm presented in Alg. 2, it is sufficient to determine the representativeness of a measurement sample based on the state sample χ which should be sampled from the propagated belief $b^-[X_{i|k+l}]$.

Algorithm 6 IsRepSample

Input:

samples	\triangleright set of candidate sampled measurements, see Alg 5 line 17
$b^-[X_{i k+l}]$	\triangleright The belief from planning time $k + l$ the samples should be representing

```

1: Given  $\beta_\sigma = 1.5$             $\triangleright$  User determined Heuristic, in direct proportion to acceptance
2: stateSamples  $\leftarrow$  the sampled states that created samples           $\triangleright$  see Alg. 2
3: for each sample  $\in$  stateSamples do
4:   if sample  $\subset \pm\beta_\sigma \cdot \sigma$  of  $b^-[X_{i|k+l}]$  then            $\triangleright$  The sample falls within  $\pm\beta_\sigma \cdot \sigma$  range
5:     {repSamples}1n  $\leftarrow$  all measurement samples  $\in$  samples that were crated by sample
6:   else                            $\triangleright$  The sample falls outside the  $\pm\beta_\sigma \cdot \sigma$  range, hence rejected
7:     {repSamples}1n  $\leftarrow$  re-sample  $n_z$  measurements using  $b^-[X_{i|k+l}]$            $\triangleright$ 
                                  freshly sampled, see Alg. 2
8:   end if
9: end for
10: return {repSamples}1n, {q(.)}1n     $\triangleright$  {q(.)}1n denote the distributions {repSamples}1n were
      sampled from

```

Considering the known (stochastic) measurement model, the space of measurement model distributions is uniquely defined by the set of state samples. So, in order to simplify the selection of representative measurement samples, we consider only the set of sampled states and assume that a set of sampled states that are representative of the propagated belief they should have been sampled

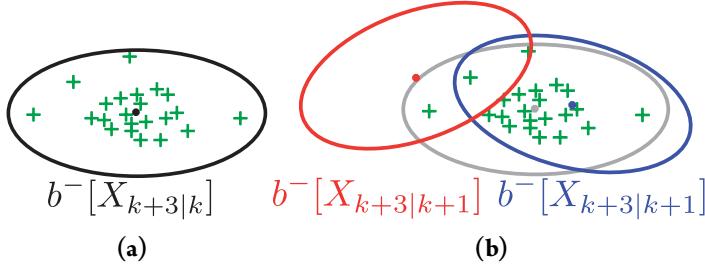


Figure 6.4.2: Illustration for adequate and inadequate representation of a belief by samples. (a) Illustrates a belief (denoted by an ellipse) over the propagated joint state at future time $k + 3$ calculated as part of planning at time k , and the twenty two samples (denoted by green "+" signs) taken from it. (b) Illustrates two instantiations of belief over the propagated joint state at future time $k + 3$ calculated as part of planning at time $k + 1$ (denoted in red and blue), overlapping the belief of precursory planning time and its samples. While the samples of (a) can be considered as adequate representation for the blue belief, they can also be considered as inadequate representation of the red belief.

from, yields a set of representative sampled measurements. Following the aforementioned, the problem of determining a set of representative measurement samples, turns into a problem of determining a set of state samples representative of some propagated belief.

Let us consider samples and $b^-[X_{i|k+l}]$ denoting respectively the candidate measurements for re-use and the propagated belief from planning time $k+l$. Due to the fact that samples were sampled from distributions different from $\mathbb{P}(z_{i+1|k+l}|H_{i+1|k+l}^-)$, we need to assure they constitute an adequate representation of it. In Alg. 6 we consider the sampled states that led to the acquired sampled measurements (Alg. 6 line 2), and denote it as stateSamples. We consider each sampled state separately, and determine sample \in stateSamples as representative if it falls within a predetermined σ range of the distribution it should have been sampled from. It is worth stressing that in order to facilitate the use of importance sampling in solving the expected reward value (as discussed later in Section 6.5) one should have access to the importance sampling distributions denoted by $\{q(\cdot)\}_1^n$ in Alg. 6.

We conclude this section with a toy example for the aforementioned method for determining a representative set of samples. Figure 6.4.2a illustrates a set of states χ (denoted by green "+" signs) used in future time $k + 3$ under planning time k , where Figure 6.4.2b illustrates how well the same samples represents two instantiations of the same future time $k + 3$ in succeeding planning at time $k + 1$. By considering some $\pm\beta_\sigma \cdot \sigma$ range of each instantiation of $b^-[X_{k+3|k+1}]$ we can determine which of the available samples can be considered as representative. Following Alg. 6, for a value of $\beta_\sigma = 1$, under the blue belief instantiation in Figure 6.4.2b, all but the left most sample will be considered as representative of $b^-[X_{k+3|k+1}]$ since they are within the covariance ellipsoid, representing the $\pm 1\sigma$ range. While under the red belief instantiation only the three samples within the red covariance ellipsoid will be considered as representative where the rest will be re-sampled from the nominal distribution.

6.4.3 BELIEF UPDATE AS PART OF IMMEDIATE REWARD CALCULATIONS

Once we determined a set of n samples we wish to use (Alg. 5 line 18), whether newly sampled, re-used or a mixture of both, we can update the appropriate beliefs in order to obtain $\{b^r[X_{i+1|k+l}]\}_{r=1}^n$, required for calculating the reward function values at the look ahead step $i + 1$. In this section we go through the belief update process, which is case sensitive to whether a sample was newly sampled or re-used. We start with the standard belief update for newly sampled measurements; continue with recalling the difference between some belief $b[X_{i+1|k+l}]$, and its counterpart from planning time k , i.e. $b[X_{i+1|k}]$; and conclude with belief update for a re-used measurement.

NEWLY SAMPLED MEASUREMENT

For a newly sampled measurement $z_{i+1|k+l}$, we follow the standard belief update of incorporating the measurement factors to the propagated belief $b^-[X_{i+1|k+l}]$ as in

$$b[X_{i+1|k+l}] \propto b^-[X_{i+1|k+l}] \cdot \prod_{j \in \mathcal{M}_{i+1|k+l}} \mathbb{P}(z_{i+1|k+l}^j | x_{i+1}, l_j), \quad (6.11)$$

and then performing inference; hence no re-use of calculations from planning time k .

RE-USED MEASUREMENT

As mentioned earlier, the motivation for re-using samples is to evert from the costly computation time of performing inference over a belief. Since we already performed inference over beliefs at planning time k , if we re-use the same samples, we can evert from performing standard belief update (6.11), and utilize beliefs from planning time k . As discussed in Section 6.1, the factors of two beliefs over the same future time but different planning sessions could be divided into three groups as illustrated in Figure 6.1.1: (i) representing shared history which is by definition identical between the two; (ii) representing potentially different factors since they are predicted for time k and given for time $k + l$; (iii) represents future time for both, but each is conditioned over different history subject to (ii), so also potentially different.

Let us consider the measurement $z_{i+1|k} \subset \{\text{repSamples}\}_1^n$, marked for re-use. The belief we are required to adjust is the one resulted from $z_{i+1|k}$ at planning time k , i.e.

$$b[X_{i+1|k}] \propto \mathbb{P}(X_{o:i+1} | H_{i+1|k}^-, z_{i+1|k}). \quad (6.12)$$

Although $b[X_{i+1|k}]$ is given to us from precursory planning, it might require an update to match the new information received up to time $k + l$. In contrary to (6.11), we update $b[X_{i+1|k}]$ using our novel

approach for inference update - RUB Inference (see Chapter 4), which enables us to incrementally update the belief solution without performing inference once more. The process of incrementally updating a belief, as thoroughly described in Section 4.6, can be divided into two general steps: first updating the DA (Section 4.6.2) and then the measurement values (Section 4.4). Let us consider the belief we wish to re-use from planning time k (6.12).

In order to update the measurement factors of (6.12) to match (6.11) we start with matching their data association (DA). As described in Alg. 1, this DA matching will provide us with the indices of the factors that their DA should be updated, as well as factors that should be added or removed. The DA update process is being done over the graphical representation of the belief, i.e. the factor graph and bayes-tree. Once the DA update is complete we are left with updating the values of all the consistent DA factors. For the special case of Gaussian beliefs, Section 4.4 provides few methods to efficiently update the aforementioned. Once the update is complete we obtain n beliefs representing $b[X_{i+1|k+l}]$, each corresponding to one of our n samples $\{\text{repSamples}\}_1^n$.

At this point it is worth reiterating the importance of ε_c (Section 6.3.3), the computational effort of updating a candidate belief is with direct correlation to the distance between the beliefs.

6.4.4 RE-USING / CALCULATING IMMEDIATE REWARD VALUES

As part of solving the planning problem (5.5) we need to get the objective value for various action sequences. The objective value for some action sequence (5.6) is given by the sum of expected rewards along the planning horizon. The expected reward value is a weighted average of immediate rewards over future belief realizations (5.14). This section deals with the calculation of those immediate rewards. Since in the general case those immediate rewards are functions of belief and action, we need to perform inference over the beliefs before we can obtain the immediate reward values. Once we have a set of beliefs representing the possible futures of executing some action u_i at future time $i + 1$ (Alg. 5 line 23), we can calculate the immediate rewards resulting from each such belief (Alg. 5 line 24).

Given a reward function for the $i + 1$ lookahead step $r_{i+1}(b, u)$, a posterior future belief $b^s[X_{i+1|k+l}]$, and the corresponding action $u_{i|k+l}$, the immediate reward $r_{i+1|k+l}^s$ is given by

$$r_{i+1|k+l}^s = r_{i+1} \left(b^s[X_{i+1|k+l}], u_{i|k+l} \right). \quad (6.13)$$

In this work, depending on the origin of $b^s[X_{i+1|k+l}]$, the immediate reward value is either calculated according to Eq. (6.13), i.e. from scratch, or being taken directly from a precursory planning session.

Under iX-BSP, any future belief is obtained by one of three ways: calculated from scratch using freshly sampled measurements (6.11); through updating a previously calculated belief with the

appropriate information; or by one of our main contributions in this work, completely re-using a previously calculated belief without any update (denoted as wildfire, Section 6.7). In this work, for the first two cases, the immediate reward value $r_{i+1|k+l}^s$ is obtained through simply solving Eq. (6.13), where for the third case the reward value $r_{i+1|k+l}^s$ is not calculated, but approximated by considering a previously calculated immediate reward value.

Between calculating the immediate reward directly from Eq. (6.13), and approximating it without any calculation based on wildfire (see Section 6.7) there is the middle ground, not used in this work and left for future work, incrementally updating a previously calculated reward value.

6.5 INCREMENTAL EXPECTATION WITH IMPORTANCE SAMPLING

This section describes one of our main contributions, incorporating Multiple Importance Sampling (MIS) into the objective estimator in order to account for selective re-use of previously calculated future beliefs. For the reader’s convenience all the notations of this section are summarized in Table 6.1. Once we obtain immediate reward values for candidate actions along the planning horizon (Alg. 3, lines 6–8), we use them to estimate (5.17). However, because we are selectively re-using samples from precursory planning sessions, we estimate (5.17) using samples not necessarily taken from $\mathbb{P}(z_{k+l+1:i|k+l}|H_{k+l|k+l}^-, u_{k+l:i-1|k+l})$, thus the formulation should be adjusted accordingly. In the following, we recall the standard general formulation for the objective function, we consider an assumption simplifying the objective weighting scheme, we then relax this simplifying assumption and characterize our problem under the MIS problem and formulate it accordingly, and finally using a simple example we demonstrate objective calculation under iX-BSP.

First let us recall the standard formulation for Eq. (5.17), for sampling $(n_x \cdot n_z)$ measurements per candidate step following Alg. 2,

$$J(u') = \sum_{i=k+l+1}^{k+l+L} \left[\eta_{k+l+1} \sum_{\{z_{k+l+1|k+l}\}} \omega_{k+l+1}^n \left(\dots \left(\eta_i \sum_{\{z_{i|k+l}\}} \omega_i^n \cdot r_i(b^n[X_{i|k+l}], u_{i-1|k+l}) \right) \dots \right) \right], \quad (6.14)$$

where ω_i^n denotes the weight of the n_{th} measurement sample for future time i , η_i denotes the normalizer of the weights at time i such that $\eta_i^{-1} = \sum_{n=1}^{n_x \cdot n_z} \omega_i^n$, and $b^n[X_{i|k+l}]$ is the belief considering a specific set of samples up to future time i , i.e.

$$b^n[X_{i|k+l}] \doteq \mathbb{P}(X_i | H_{k+l|k+l}, u_{k+l:i-1|k+l}, \{z_{k+l+1|k+l}\}, \dots, \{z_{i|k+l}\}). \quad (6.15)$$

In iX-BSP we are potentially forcing samples from previous planning sessions; this type of prob-

lems that involve expressing one distribution using samples taken from another is referred to as importance sampling (see Appendix D). It is worth stressing that unlike [50] which uses importance sampling to sample sets of future states actions and measurements, we make use of importance sampling to incorporate re-used beliefs within the objective estimation.

Let us temporarily consider an assumption that would simplify the weighting scheme required for iX-BSP under importance sampling. Assume we are at planning time $k + l$, considering planning session at time k for re-use under iX-BSP. At each lookahead step $i \in [k+l+1 : k+L]$ we conclude that all candidate measurements from planning at time k form a representative set of $\mathbb{P}(z_{i|k+l} | H_{i|k+l})$, so we decide to re-use them all. Moreover, all of these samples from lookahead step i at planning time k were sampled from the same distribution denoted as $q_{i|k}(\cdot)$, whether they were all freshly sampled at planning time k or have been entirely re-used themselves from some past time in which they were freshly sampled. Under this scenario, we have a set of representative measurements, all sampled from the same distribution that is not the nominal one, thus Eq. (6.14) can be written as

$$J(u') \sim \sum_{i=k+l+1}^{k+l+L} \left[\frac{1}{n_i} \sum_{g=1}^{n_i} \omega_i(z_{k+l+1:i}^g) \cdot r_i(b^g[X_{i|k+l}], u'_{i-1|k+l}) \right], \quad (6.16)$$

where n_i denotes the number of samples used in the $i - k - l$ lookahead step, and the weights are given by a simple probability ratio, which under importance sampling is usually referred to as the likelihood ratio (see Appendix D),

$$\omega_i(z_{k+l+1:i}^g) = \frac{\mathbb{P}(z_{k+l+1:i}^g | H_{k+l|k+l}, u_{k+l:i-1|k+l})}{q_i(z_{k+l+1:i}^g)}, \quad (6.17)$$

where $\mathbb{P}(\cdot)$ denotes the measurement likelihood we should have sampled the measurements from, and $q_i(\cdot)$ denotes the probability distribution we actually sampled from. For example, assuming we re-used measurements from planning at time k , the likelihood ratio will be given by

$$\omega_i(z_{k+l+1:i}^g) = \frac{\mathbb{P}(z_{k+l+1:i}^g | H_{k+l|k+l}, u_{k+l:i-1|k+l})}{\mathbb{P}(z_{k+l+1:i}^g | H_{k+l|k}, u_{k+l:i-1|k})}. \quad (6.18)$$

While this formulation under the simplified assumption allows one to easily re-use previous planning sessions, it is only under an "all or nothing" approach, which cripples the full potential of iX-BSP for selective re-use of previous planning sessions. Our problem as part of iX-BSP is more specific: at each look-ahead step we can potentially force samples from M different measurement likelihood distributions (e.g. Section 6.5.1), which none of them is necessarily the nominal one. The possible

number of distributions from which measurements are being sampled in every look-ahead step i is bounded by

$$1 \leq M \leq n_i. \quad (6.19)$$

The lower bound in (6.19) would occur either when there are no re-used samples, or when all samples are re-used and were originally sampled from the same distribution e.g. from a specific measurement likelihood in a previous planning session, as in (6.18). The upper bound in (6.19) would occur when $n_m = 1 \forall m$ i.e. each sample has been obtained using a different distribution, e.g. from m different measurement likelihoods, potentially, from different planning sessions.

This problem falls within the multiple importance sampling problem (see Appendix D), and as such we can reformulate the estimator for (5.17) accordingly

$$J(u') \sim \sum_{i=k+l+1}^{k+l+L} \left[\sum_{m=1}^{M_i} \frac{1}{n_m} \sum_{g=1}^{n_m} \tilde{\omega}_m(z_{k+l+1:i}^{m,g}) \cdot \frac{\mathbb{P}(z_{k+l+1:i}^{m,g} | H_{k+l|k+l}, u_{k+l:i-1|k+l})}{q_m(z_{k+l+1:i}^{m,g})} \cdot r_i(b^{m,g}[X_{i|k+l}], u'_{i-1|k+l}) \right], \quad (6.20)$$

where i denotes the look-ahead step, M_i is the number of distributions from which measurements are being sampled, n_m is the number of measurements sampled from the m_{th} distribution, $q_m(\cdot)$ is the m_{th} distribution that samples were taken from, and $z_{k+l+1:i}^{m,g}$ are the g_{th} set of future measurements at time instances $k + l + 1 : i$, sampled from the m_{th} distribution. The m_{th} weight is denoted by $\tilde{\omega}_m$ where $\sum \tilde{\omega}_m = 1$, and $\tilde{\omega}_m > 0 \forall m$. The estimator (6.20) is unbiased under the assumption that $q_m(\cdot) > 0$ whenever $\tilde{\omega}_m(\cdot) \cdot \mathbb{P}(z|\mathcal{H}) \cdot r_i(\cdot) \neq 0$. When using previous planning sessions as candidates for re-use under iX-BSP, $q_m(\cdot)$ corresponds to a measurement likelihood of those previous planning sessions (e.g. Section 6.5.1).

In this work we made use of the unbiased nearly optimal estimator for (5.17), based on the multiple importance sampling problem with the balance heuristic (see Appendix D)

$$J(u') \sim \sum_{i=k+l+1}^{k+l+L} \left[\frac{1}{n_i} \sum_{m=1}^{M_i} \sum_{g=1}^{n_m} \omega_i(z_{k+l+1:i}^{m,g}) \cdot r_i(b^{m,g}[X_{i|k+l}], u'_{i-1|k+l}) \right], \quad (6.21)$$

where i denotes the look-ahead step, n_i are the number of samples considered, M_i is the number of distributions from which measurements are being sampled, n_m is the number of measurements sampled from the m_{th} distribution, and following the balance heuristic $\omega_i(z^{m,g})$ is the likelihood ratio of

Variable	Description
$\square_{t k}$	Of time t while current time is k
$b[X_{t k}]$	belief at time t while current time is k
$b^n[X_{t k+l}]$	The n_{th} sampled belief representing $b[X_{t k+l}]$
ω_i^n	the weight corresponding to the n_{th} measurement sample for lookahead step i
$q_i(z_{t:i}^g)$	importance sampling distribution at lookahead step i , from which $z_{t:i}^g$ were sampled
$\mathbb{P}(z_{t:i}^\square H, u)$	the nominal distribution at lookahead steps $t : i$
n_i	the number of samples considered at lookahead step i
M_i	number of distributions at look ahead step i from which measurements are being sampled
n_m	the number of measurements sampled from the m_{th} distribution at look ahead step i
$\omega_i(z_{t:i}^g)$	private case of $\omega_i(z_{t:i}^{m,g})$ where $m = 1$
$\omega_i(z_{t:i}^{m,g})$	Balance Heuristic likelihood ratio at lookahead step i corresponding to $z_{t:i}^{m,g}$
$z_{t:i}^{m,g}$	the g_{th} set of future measurements at time instances $t : i$ sampled from the m_{th} distribution
$b^{m,g}[X_{t k+l}]$	the sampled belief representing $b^{m,g}[X_{t k+l}]$ which consider the measurements $z_{k+l+1:t}^{m,g}$
$q_m()$	the m_{th} marginal importance sampling distribution at lookahead step i , $m \in [1, M_i]$
$b^-[X_{t k}]$	belief at time $t - 1$ propagated only with action $u_{t-1 k}$
$\tilde{b}[X_{t k}]$	The root of the selected branch for re-use from planning time k
$\mathcal{B}_{t k}$	The set of all beliefs from planning time k rooted in $\tilde{b}[X_{t k}]$
$\{b[X_{t k}]\}_1^j$	j sampled beliefs representing $b[X_{t k}]$
$\{r_i(b[X_{t k}], u)\}_1^j$	j immediate rewards of lookahead step i
$p_i()$	the marginal nominal distribution at lookahead step i
$\tilde{p}_i()$	the nominal distribution at lookahead step i
$\tilde{q}_m()$	the m_{th} importance sampling distribution at lookahead step i , $m \in [1, M_i]$
$\mathbb{D}(\cdot)$	belief divergence / metric

Table 6.1: Notations for Section 6.5

the g_{th} sample from the m_{th} distribution given by

$$\omega_i(z_{k+l+1:i}^{m,g}) = \frac{\mathbb{P}(z_{k+l+1:i}^{m,g} | H_{k+l|k+l}, u_{k+l:i-1|k+l})}{\sum_{\tilde{m}=1}^{M_i} \frac{n_{\tilde{m}}}{n_i} q_{\tilde{m}}(z_{k+l+1:i}^{m,g})}, \quad (6.22)$$

where $z_{k+l+1:i}^{m,g}$ are the g_{th} set of future measurements at time instances $[k+l+1 : i]$, sampled from the m_{th} distribution, and $q_{\tilde{m}}(\cdot)$ is the \tilde{m}_{th} importance sampling distribution.

The balance heuristic is considered as nearly optimal in the following sense:

Lemma 1. Let $n_m \geq 1$ be positive integers for $m = 1, \dots, M_i$. Let $\tilde{\omega}_1, \dots, \tilde{\omega}_{M_i}$ be a partition of unity and let ω^{BH} be the balance heuristic. Let $\tilde{J}_{\tilde{\omega}_m}$ and $\tilde{J}_{\omega^{BH}}$ be the estimates of J under $\tilde{\omega}_m$ and ω^{BH} respectively. Then

$$Var(\tilde{J}_{\omega^{BH}}) \leq Var(\tilde{J}_{\tilde{\omega}_m}) + \left(\frac{1}{\min_m n_m} - \frac{1}{\sum_m n_m} \right) J^2 \quad (6.23)$$

Proof. This is Theorem 1. of [74]. □

When all samples being considered to estimate (5.17) are sampled from their nominal distributions, (6.21) is reduced back to Eq. (5.15), with all the weights degenerating to ones; for such a case, $M_i = 1 \forall i$, and $q_1(\cdot) = \mathbb{P}(z_{k+l+1:i} | H_{k+l|k+l}, u_{k+l:i-1|k+l})$, thus $\omega_i = 1 \forall i$. When for each lookahead step, all samples share the same distribution that is not the nominal one, i.e. $M_i = 1 \forall i$, and $p(\cdot) \neq q(\cdot)$, thus Eq. (6.21) is reduced back to Eq. (6.16).

6.5.1 iX-BSP WALK-THROUGH EXAMPLE

To better understand the objective value calculation under iX-BSP, let us perform iX-BSP over a simple example. Assume we have access to all calculations from planning time k , in-which we performed X-BSP (or iX-BSP) for a horizon of three steps, and with $n_x = 2$ and $n_z = 1$. Figure 6.5.1a illustrates a specific action sequence, $u_1 \rightarrow u_2 \rightarrow u_1$, considered as part of planning at time k . Let us assume that the optimal action decided upon as part of planning at time k , and was later executed was u_1 . We are currently at time $k+1$, after performing inference using the measurements we received as a result of executing u_1 . We perform planning using iX-BSP with the same horizon length and number of samples per action, for several action sequences, one of which is the action sequence $u_2 \rightarrow u_1 \rightarrow u_2$, as illustrated in Figure 6.5.1c.

Following Alg. 3 line 1, out of the two available beliefs from planning time k shown in Figure 6.5.1a, $\{b[X_{k+1|k}]\}_1^2$, the left one is determined as closer to $b[X_{k+1|k+1}]$, so we consider all its descendants as

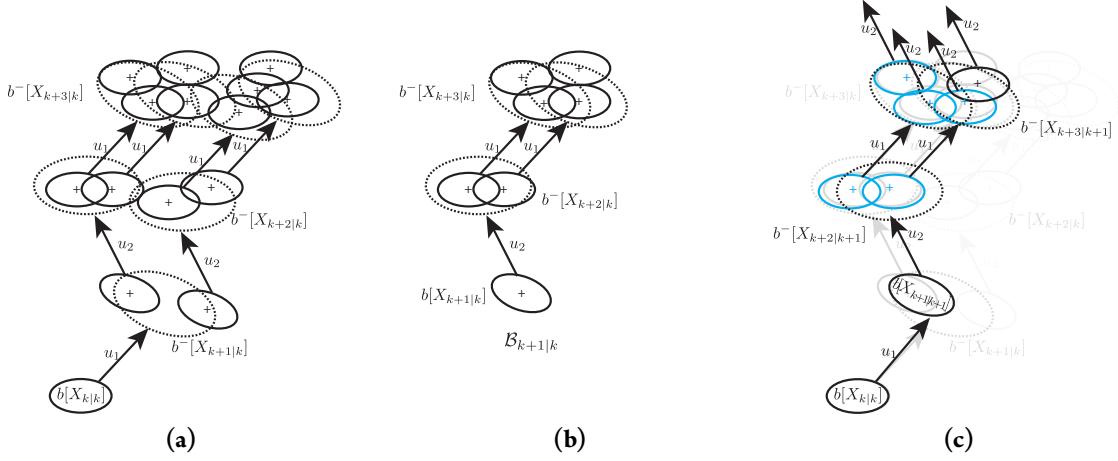


Figure 6.5.1: Illustration of two consecutive planning sessions, each with horizon of three steps, and two sampled measurements per action. Beliefs are denoted by solid ellipses, propagated beliefs by dotted ellipses, and beliefs that have been re-used are denoted in blue.
(a) Planning over the action sequence $u_1 \rightarrow u_2 \rightarrow u_1$ at time k (b) The selected branch for re-use $\mathcal{B}_{k+1|k}$ from planning at time k (c) The segment of planning at time $k+1$ for the action sequence $u_2 \rightarrow u_1 \rightarrow u_2$ that overlaps with (b).

the set $\mathcal{B}_{k+1|k}$, as illustrated in Figure 6.5.1b, and define Dist as

$$\text{Dist} \triangleq \mathbb{D}(b[X_{k+1|k+1}], b[X_{k+1|k}]), \quad (6.24)$$

where $\mathbb{D}(\cdot)$ denotes some belief distance. For the sake of this example let us say Dist is determined as close enough for re-use; we can therefore continue with re-using the beliefs in the set $\mathcal{B}_{k+1|k}$ (Alg. 3 line 6). First we check which of the two available sampled measurements from planning time k constitutes an adequate representation for $\mathbb{P}(z_{k+2}|H_{k+1|k+1}, u_2)$. One way to do so, is following Alg. 6 and checking whether the two available state samples from planning time k constitute an adequate representation for $b^-[X_{k+2|k+1}]$; since they are, we consider all measurements associated to them as a representative set of $\mathbb{P}(z_{k+2}|H_{k+1|k+1}, u_2)$. Our representative set of measurement samples for look ahead step $k+2$ now holds two re-used measurements, so we update their corresponding beliefs $\{b[X_{k+2|k}]\}_1^2$ into $\{b[X_{k+2|k+1}]\}_1^2$ (Alg. 5 line 23), the updated beliefs are denoted in blue in Figure 6.5.1c. After updating the beliefs we can calculate/ update the immediate rewards(costs) associated with them, see Section 6.4.4, once obtained we can proceed to the next future time step.

For the next look ahead step, we propagate $\{b[X_{k+2|k+1}]\}_1^2$ with action u_1 to obtain $\{b^-[X_{k+3|k+1}]\}_1^2$ (Alg. 5 line 9), and check whether the four available measurement samples from planning time k constitute an adequate representation for $\mathbb{P}(z_{k+3}|H_{k+2|k+1}, u_1)$ (Alg. 5 line 18); following Alg. 6 we find only three of them are, so we mark the associated beliefs for re-use, and sample the forth measurement from the original distribution $\mathbb{P}(z_{k+3}|H_{k+2|k+1}, u_1)$. We then update the beliefs we marked for re-use, $\{b[X_{k+3|k}]\}_1^3$ into $\{b[X_{k+3|k+1}]\}_1^3$ (denoted by the blue colored beliefs at $k+3|k+1$ in Figure 6.5.1c),

and $b^-[X_{k+3|k+1}]$ into $b^4[X_{k+3|k+1}]$ (denoted by the black colored belief at $k+3|k+1$ in Figure 6.5.1c) using the newly sampled measurement (Alg. 5 line 23). After obtaining the beliefs for look ahead step $k+3$, whether through updating a re-used belief or calculation from scratch, we calculate/ update the immediate rewards(costs) of each. Since we do not have candidate beliefs to be re-used for the next time step, the last step of the horizon $k+4|k+1$ is calculated using X-BSP (Alg. 3 line 8).

At this point we have all the immediate rewards for each of the predicted beliefs along the action sequence $u_2 \rightarrow u_1 \rightarrow u_2$, so we can calculate the expected reward value for this action sequence for planning at time $k+1$. For the look ahead step $k+2$ of planning session at time $k+1$, i.e. $k+2|k+1$, we have two reward values, $\{r_{k+2|k+1}(b[X_{k+2|k+1}], u_2)\}_1^2$, each calculated for a different belief $b[X_{k+2|k+1}]$ considering a different sample $z_{k+2|k}$. Calculating the expected reward value for future time step $k+2|k+1$ would mean in this case, using measurements sampled from $\mathbb{P}(z_{k+2}|H_{k+1|k}, u_2)$ rather than from $\mathbb{P}(z_{k+2}|H_{k+1|k+1}, u_2)$. The use of Multiple Importance Sampling (MIS) enables us to calculate expectation while sampling from a mixture of probabilities, where the balance heuristic is used to calculate the weight functions. Using the formulation of MIS along with the balance heuristic presented in Eq. (6.21), we can write down the estimation for the expected reward value of look ahead step $k+2|k+1$,

$$\mathbb{E}[r_{k+2|k+1}(\cdot)] \sim \frac{1}{2} \frac{p_1(z_{k+2|k}^{1,1})}{\frac{3}{2}q_1(z_{k+2|k}^{1,1})} \cdot r_{k+2|k+1}^1(\cdot) + \frac{1}{2} \frac{p_1(z_{k+2|k}^{2,1})}{\frac{3}{2}q_1(z_{k+2|k}^{2,1})} \cdot r_{k+2|k+1}^2(\cdot), \quad (6.25)$$

where $p_1(\cdot) \doteq \mathbb{P}(z_{k+2}|H_{k+1|k+1}, u_2)$ and $q_1(\cdot) \doteq \mathbb{P}(z_{k+2}|H_{k+1|k}, u_2)$. In the same manner, following (6.21), we can also write down the estimation for the expected reward value at look ahead step $k+3$ of planning at time $k+1$, i.e. $k+3|k+1$,

$$\begin{aligned} \mathbb{E}[r_{k+3|k+1}(\cdot)] &\sim \frac{1}{4} \frac{\tilde{p}_2(z_{k+2:k+3|k}^{1,1})}{\frac{3}{4}\tilde{q}_2(z_{k+2:k+3|k}^{1,1}) + \frac{1}{4}\tilde{p}_2(z_{k+2:k+3|k}^{1,1})} r_{k+3|k+1}^1(\cdot) \\ &+ \frac{1}{4} \frac{\tilde{p}_2(z_{k+2:k+3|k}^{2,1})}{\frac{3}{4}\tilde{q}_2(z_{k+2:k+3|k}^{2,1}) + \frac{1}{4}\tilde{p}_2(z_{k+2:k+3|k}^{2,1})} r_{k+3|k+1}^2(\cdot) \\ &+ \frac{1}{4} \frac{\tilde{p}_2(z_{k+2:k+3|k}^{3,1})}{\frac{3}{4}\tilde{q}_2(z_{k+2:k+3|k}^{3,1}) + \frac{1}{4}\tilde{p}_2(z_{k+2:k+3|k}^{3,1})} r_{k+3|k+1}^3(\cdot) \\ &+ \frac{1}{4} \frac{\tilde{p}_2(z_{k+2:k+3|k}^{4,1})}{\frac{3}{4}\tilde{q}_2(z_{k+2:k+3|k}^{4,1}) + \frac{1}{4}\tilde{p}_2(z_{k+2:k+3|k}^{4,1})} r_{k+3|k+1}^4(\cdot), \end{aligned} \quad (6.26)$$

where $\tilde{p}_2(\cdot) \doteq \mathbb{P}(z_{k+2:k+3}|H_{k+1|k+1}, u_2, u_1)$ and $\tilde{q}_2(\cdot) \doteq \mathbb{P}(z_{k+2:k+3}|H_{k+1|k}, u_2, u_1)$. When considering (5.9), we can re-write the measurement likelihood from (6.26) into a product of measurement

likelihoods per look ahead step, e.g. $\tilde{p}_2(z_{k+2:k+3|k}^1) = p_1(z_{k+2|k}^1)p_2(z_{k+3|k}^1)$,

$$\begin{aligned}\mathbb{E}[r_{k+3|k+1}(.)] &\sim \frac{1}{4} \frac{p_1(z_{k+2|k}^{1,1})p_2(z_{k+3|k}^{1,1})}{\frac{3}{4}q_1(z_{k+2|k}^{1,1})q_2(z_{k+3|k}^{1,1}) + \frac{1}{4}p_1(z_{k+2|k}^{1,1})p_2(z_{k+3|k}^{1,1})} r_{k+3|k+1}^1(.) \\ &+ \frac{1}{4} \frac{p_1(z_{k+2|k}^{1,1})p_2(z_{k+3|k}^{2,1})}{\frac{3}{4}q_1(z_{k+2|k}^{1,1})q_2(z_{k+3|k}^{2,1}) + \frac{1}{4}p_1(z_{k+2|k}^{1,1})p_2(z_{k+3|k}^{2,1})} r_{k+3|k+1}^2(.) \\ &+ \frac{1}{4} \frac{p_1(z_{k+2|k}^{2,1})p_2(z_{k+3|k}^{3,1})}{\frac{3}{4}q_1(z_{k+2|k}^{2,1})q_2(z_{k+3|k}^{3,1}) + \frac{1}{4}p_1(z_{k+2|k}^{2,1})p_2(z_{k+3|k}^{3,1})} r_{k+3|k+1}^3(.) \\ &+ \frac{1}{4} \frac{p_1(z_{k+2|k}^{2,1})p_2(z_{k+3|k}^{4,1})}{\frac{3}{4}q_1(z_{k+2|k}^{2,1})q_2(z_{k+3|k}^{4,1}) + \frac{1}{4}p_1(z_{k+2|k}^{2,1})p_2(z_{k+3|k}^{4,1})} r_{k+3|k+1}^4(.)\end{aligned}\quad (6.27)$$

where $p_1(.)$ need not be calculated at look ahead step $k + 3$, since it is already given from (6.25).

6.6 RESULTS - IX-BSP

In order to examine the effect of calculation re-use under the iX-BSP paradigm, we compare the runtime of iX-BSP and X-BSP using Active full SLAM as a test-bed under Model Predictive Control (MPC) framework. To better understand the differences between X-BSP and iX-BSP, let us consider them inside a plan-act-infer system. Figure 6.6.1a illustrates the high level algorithm for plan-act-infer using X-BSP, marking the section of the algorithm which is being timed for comparison. Figure 6.6.1b illustrates the high level algorithm for plan-act-infer using iX-BSP, marking the section of the algorithm which is timed for comparison. As can easily be seen in Figure 6.6.1 all differences between X-BSP and iX-BSP are confined within the planning block, hence the computation time of the planning process is adequate for fair comparison. It is important to mention that no offline calculations whatsoever, are involved in any of the comparisons. For simplicity all results consider known motion and observation models with zero mean Gaussian noise as well as motion primitives. Both X-BSP and iX-BSP consider a planning horizon of 3 steps, 3 candidate actions (forward, left and right), with all the possible permutations between them - hence 27 candidate action sequences, $n_x = 5 n_z = 1, 6$ DOF robot pose, 3 DOF landmarks and a joint state comprised of both robot poses and landmarks. In the following we provide a statistical comparison between X-BSP and iX-BSP (Section 6.6.1) under a simplifying assumption that all previously sampled measurements can be re-used in current planning time, and a statistical comparison between X-BSP and iX-BSP with selective re-sampling (Section 6.6.2).

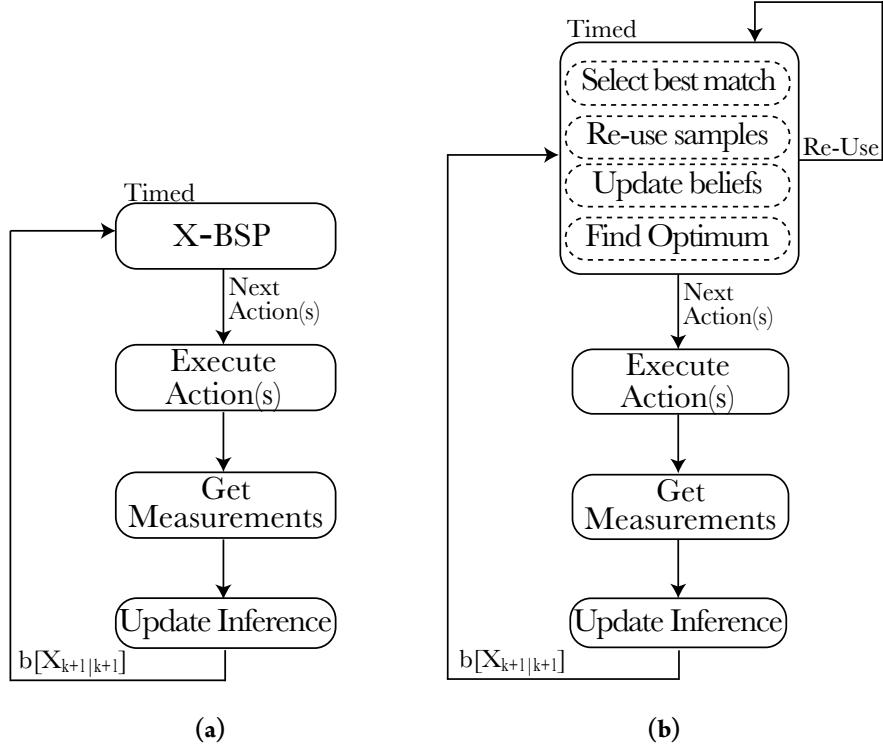


Figure 6.6.1: High level algorithm for iX-BSP presented in a block diagram: (a) presents a standard plan-act-infer framework with Bayesian inference and X-BSP; (b) presents our novel approach for incremental expectation BSP - iX-BSP. Instead of calculating planning from scratch we propose to utilize the precursory planning session.

6.6.1 RE-USING THE PRECURSORY PLANNING SESSION

In this section we examine iX-BSP with the \mathbb{D}_{DA} divergence, without wildfire and under the simplifying assumption that previously sampled measurements always constitute an adequate representation of the measurement likelihood, i.e. once the closest belief is selected using \mathbb{D}_{DA} , all associated previously sampled measurements are re-used (see parameters in Table 6.1).

Moreover we continue the point made in Section 5.5 and run ML-BSP alongside for comparison.

To that end we compare ML-BSP, X-BSP and iX-BSP in the sense of planning-session computation time and the posterior estimation error upon reaching the goal. For comparison we perform 100 rollouts (entire mission run), each with a different sampled ground-truth for the prior state. For each rollout, we time the planning sessions of all three methods. Code implemented in MATLAB using iSAM2 efficient methodologies, and executed on a MacBookPro 2017, with 2.9GHz Intel Core i7 processor and 16GB of RAM. Figure 6.6.2a presents the scenario on which all rollouts were performed. Considering the same world and same landmarks as in Section 5.5. A robot equipped with

Prior belief standard deviation	$\begin{bmatrix} 1^o \cdot I_{3 \times 3} & \text{o} \\ \text{o} & 5[m] \cdot I_{3 \times 3} \end{bmatrix}$
Motion Model standard deviation	$\begin{bmatrix} 0.5^o \cdot I_{3 \times 3} & \text{o} \\ \text{o} & 0.5[m] \cdot I_{3 \times 3} \end{bmatrix}$
Observation Model standard deviation	$\begin{bmatrix} 3[px] & \text{o} \\ \text{o} & 3[px] \end{bmatrix}$
Camera Aperture	90^o
Camera acceptable Sensing Range useWF	between $2[m]$ and $40[m]$ false
ε_c	250
β_σ	∞
n_u	3
n_x	5
n_z	1
action primitives	left, right and forward with $1[m]$ translation and $\pm 90^o$ rotations
\mathbb{D}	\mathbb{D}_{DA}

Table 6.1: Parameters for Section 6.6.1 following Alg. 3

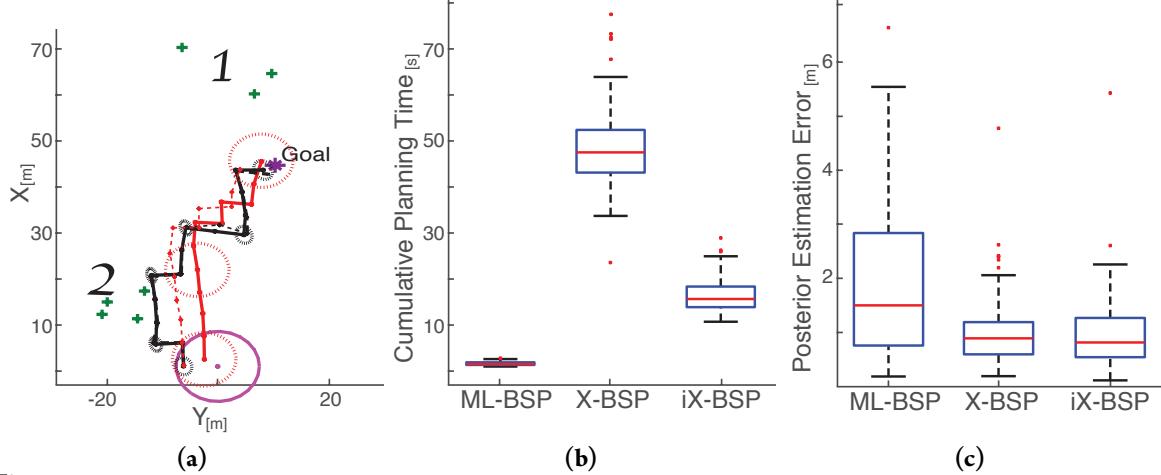


Figure 6.6.2: (a) Testing scenario, landmarks denoted by green "+" , prior state and uncertainty in solid purple, ML-BSP denoted by red, X-BSP and iX-BSP denoted by black. (b) and (c) are Box plots of 100 rollouts for planning session timing (b) and posterior estimation error (c) upon reaching the goal.

a stereo camera, is required to reach the goal whilst not crossing a covariance threshold, i.e. cost consisting of distance to goal and a covariance penalty above a certain value. Figure 6.6.2a shows one of the 100 rollouts that were calculated, in which the estimated trajectory by each method is denoted by a solid line, the ground truth by a dashed line and the posterior covariance by a dashed ellipse. In Figure 6.6.2a both X-BSP and iX-BSP, in black, chose the same optimal actions along the mis-

sion, while ML-BSP, in red, chose differently. We can also see the effect of this difference over each method's covariance, X-BSP and iX-BSP action choice led to a smaller covariance along the entire path.

Figure 6.6.2b presents the statistical data of the planning session running time. Since in this example we follow an MPC framework, the last step of each horizon is required to be calculated from scratch. Since doing so is identical to the course of action in X-BSP, we present the computation time of the entire horizon, excluding the last horizon step. As expected, for average timing as well as for each separate rollout, both ML-BSP and iX-BSP timings are lower than that of X-BSP. By reusing previous planning session, instead of calculating it from scratch we save valuable computation time, theoretically without effecting the planning solution. We examine the effect on the planning solution in Figure 6.6.2c, by comparing the posterior estimation error upon reaching the goal. As expected, the statistical results of 100 rollouts presented in Figure 6.6.2c, shows that X-BSP is statistically superior to ML-BSP: in 63% of the rollouts it has a smaller estimation error while in 10% they are equal. Importantly, we can also see that iX-BSP is statistically similar to X-BSP, with 41% of the rollouts with smaller estimation error and 15% equal. We note that relaxing the simplifying assumption that all samples are adequately representative, would result with an even better match between X-BSP and iX-BSP.

6.6.2 iX-BSP

In this section we examine iX-BSP with the $\mathbb{D}_{\sqrt{J}}$ distance, without wildfire and without the simplifying assumptions used in Section 6.6.1 (see parameters in Table 6.2).

We compare iX-BSP and X-BSP in the sense of planning-session computation time, the posterior estimation error upon reaching the goal, and the covariance norm upon reaching the goal. Code implemented in MATLAB using iSAM2 efficient methodologies and executed on a Linux machine, with Xeon E3-1241v3 3.5GHz processor with 64GB of memory. For comparison we perform 20 rollouts (entire mission run), each with a different sampled ground-truth for the prior state, on 10 different, randomly generated, maps presented in Figure 6.6.3a. Each map, contains two goals and between 2 to 150 landmarks. The goals and landmarks location as well as the number of landmarks are all randomly generated for each map. For each of the 200 rollouts, we clock the planning session computation time of both methods for comparison. Across all randomized maps, the robot, equipped with a stereo camera, has the same mission - reaching each one of the goals whilst maximizing information gain and minimizing distance to goal using the reward function

$$r_i() = \alpha \cdot \frac{1}{2} \ln [(2\pi e)^n \cdot \det(\Lambda_i)] + (1 - \alpha) \cdot (D_2 G_{i-1} - D_2 G_i), \quad (6.28)$$

Prior belief standard deviation	$\begin{bmatrix} 1^\circ \cdot I_{3x3} & \text{o} \\ \text{o} & S[m] \cdot I_{3x3} \end{bmatrix}$
Motion Model standard deviation	$\begin{bmatrix} \text{o}.S^\sigma \cdot I_{3x3} & \text{o} \\ \text{o} & \text{o}.S[m] \cdot I_{3x3} \end{bmatrix}$
Observation Model standard deviation	$\begin{bmatrix} 3[\text{px}] & \text{o} \\ \text{o} & 3[\text{px}] \end{bmatrix}$
Camera Aperture	90°
Camera acceptable Sensing Range useWF	between $2[m]$ and $4o[m]$ false
ε_c	250
β_σ	1.5
n_u	3
n_x	5
n_z	1
action primitives	left, right and forward with $1[m]$ translation and $\pm 90^\circ$ rotations
\mathbb{D}	$\mathbb{D}_{\sqrt{J}}$

Table 6.2: Parameters for Section 6.6.2 following Alg. 3

where $\alpha \in [0, 1]$ is a weighting parameter, n represents the dimension of the robot's pose, Λ_i represents the focused information matrix at time i , D_2G_i represents distance-to-goal at time i .

Figure 6.6.3b presents a box-plot for the timing data of all 200 rollouts, each with 6 outliers, where the computation time advantage in favor of iX-BSP is easily noticed. The significant reduction in computation time is originated in the fact that iX-BSP performs inference update in a more efficient way, computation wise, compared to X-BSP. By forcing previously sampled measurements as part of the objective estimation, iX-BSP is able to utilize previously solved beliefs from a precursory planning session, and efficiently update them, instead of performing inference from scratch as done in X-BSP.

Since we claim to provide a more efficient paradigm to the general problem of X-BSP, we also verify how iX-BSP favors in estimation results. Figure 6.6.3c presents a box plot of the estimation error upon reaching the goal for each of the methods. The estimation error was calculated using the normalized distance between the last pose estimation and the last pose ground truth value, i.e.

$$\text{Estimation err} = \| \hat{x}_{final} - x_{final}^{gt} \| . \quad (6.29)$$

As can be seen in Figure 6.6.3c both methods average around an estimation error of $2[m]$, while X-BSP

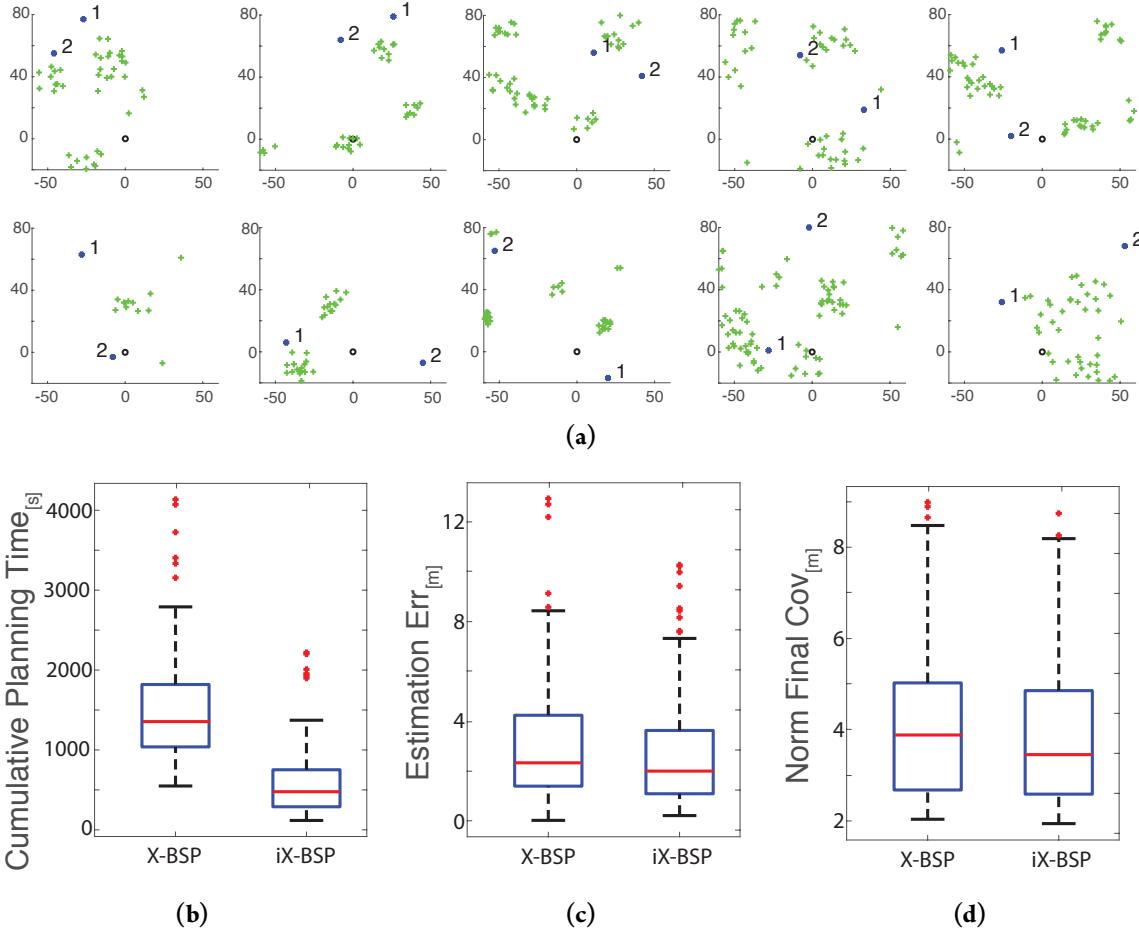


Figure 6.6.3: Statistical comparison between X-BSP and iX-BSP:(a) 10 randomly generated maps, used for the statistical comparison. Each with 2 numbered goals denoted by blue dots, and between 2 to 150 landmarks denoted by green crosses. (b) (c) and (d) Box plots of 20 rollouts for planning session timing (b), posterior estimation error upon reaching the goal (c) and the covariance norm upon reaching the goal.

is with 5 outliers and iX-BSP is with 9 outliers. In 49.5% of the rollouts, iX-BSP provided with a better estimation error than X-BSP. The large variance that can be seen in Figure 6.6.3c is probably the result of using a small number of samples for estimating the objective. Nonetheless the empiric estimation variance of both methods can be considered as statistically identical for all practical purposes. Of course a more rigorous examination is required, by analytically comparing the estimation variance, we leave this for future work. We push further and compare the covariance norm of the final pose. As can be seen in Figure 6.6.3d, they average around $3.4[m]$ for iX-BSP and $3.8[m]$ for X-BSP, with only 2 and 3 outliers respectively, and can be considered as statistically identical for all practical purposes. As suggested in Section 6.6.1, relaxing the simplifying assumption of adequately representative samples, in fact resulted with a better match between X-BSP and iX-BSP as can be seen from comparing Figure 6.6.2c to Figure 6.6.3c.

6.7 THE WILDFIRE ASSUMPTION

One of our main contributions in this chapter is the wildfire condition, which allows one to sacrifice statistical accuracy in favor of reduction in computation time by re-using an entire subtree from a previous planning session without any update. As explained in the opening of Chapter 6, at each lookahead step instead of calculating the predicted beliefs from scratch, we locate and incrementally update some previously calculated beliefs which are close enough. This belief update is more efficient, in terms of computation time (see Chapter 4), than performing inference from scratch over future measurements, hence the computation time advantage of iX-BSP over X-BSP. The wildfire condition allows to refrain even from updating the belief, thus sacrificing statistical accuracy in favor of further reduction in computation time of the planning process under iX-BSP.

The wildfire condition is a distance condition between two beliefs, used to check if one is close to the other up to some predetermined value ε_{wf} . When the condition is met, the entire subtree rooted in the considered belief is re-used "as is" without any additional update. Under our problem of iX-BSP, we consider previously calculated beliefs from planning at time k for re-use at time $k+l$ by checking the information gap between the two planning times for each planning horizon $i \in [k+l, k+L]$.

This section covers the wildfire condition, starting with the intuition behind it and its working principle (Section 6.7.1), how it is integrated within iX-BSP (Section 6.7.2), and concluding with formulating bounds for the objective value error under the use of wildfire (Section 6.7.3). For the reader's convenience all the notations of this section are summarized in Table 6.1.

6.7.1 INTUITION AND WORKING PRINCIPLE

Let us assume we found a belief from a precursory planning session $b[X_{i+1|k}] \in \mathcal{B}_{k+l|k}$ that is identical to the belief we would like to calculate $b^s[X_{i+1|k+l}]$; these beliefs would, of course, yield zero distance,

$$\mathbb{D}(b[X_{i+1|k}], b^s[X_{i+1|k+l}]) = 0. \quad (6.30)$$

For this case, instead of solving Eq. (6.13) in order to obtain $r_{i+1|k+l}^s$, we can simply use the previously calculated immediate reward associated with $b[X_{i+1|k}]$

$$r_{i+1|k+l}^s = r_{i+1} (b^s[X_{i+1|k+l}], u_{i|k+l}) \equiv r_{i+1} (b[X_{i+1|k}], u_{i|k}), \quad (6.31)$$

not only we re-use this immediate reward but we can simply re-use the entire sub-tree rooted in $b[X_{i+1|k}]$ as is. Under the wildfire condition we consider an approximation to the immediate reward value $r_{i+1|k+l}^s$ by using the immediate reward value of a previously calculated belief $b[X_{i+1|k}]$ which is

ε_{wf} close to $b^s[X_{i+1|k+l}]$ in the $\mathbb{D}(.)$ sense, i.e.

$$r_{i+1|k+l}^s = r_{i+1}(b^s[X_{i+1|k+l}], u_{i|k+l}) \cong r_{i+1}(b[X_{i+1|k}], u_{i|k}), \quad (6.32)$$

where

$$\mathbb{D}(b[X_{i+1|k}], b^s[X_{i+1|k+l}]) \leq \varepsilon_{wf}. \quad (6.33)$$

As such the immediate reward approximation error under the use of wildfire is given by

$$err_r = |r_{i+1}(b^s[X_{i+1|k+l}], u_{i|k+l}) - r_{i+1}(b[X_{i+1|k}], u_{i|k})| \quad (6.34)$$

and as claimed in Theorem 1, it can be bounded by our choice of ε_{wf} .

Theorem 1 (Bounded reward difference). *Let $r(b, u)$ be α -Hölder continuous with λ_α and $\alpha \in (0, 1]$. Let b and b' denote two beliefs. Then the difference between $r(b, u)$ and $r(b', u)$ is bounded by*

$$|r(b, u) - r(b', u)| \leq (4 \cdot \ln 2)^{\frac{\alpha}{2}} \cdot \lambda_\alpha \cdot \mathbb{D}_{\sqrt{J}}(b, b'), \quad (6.35)$$

where

$$\mathbb{D}_{\sqrt{J}}(b, b') = \sqrt{\frac{1}{2}\mathbb{D}_{KL}(b||b') + \frac{1}{2}\mathbb{D}_{KL}(b'||b)}, \quad (6.36)$$

and $\mathbb{D}_{KL}(.)$ is the KL divergence.

Proof. See Appendix F. □

Based on Theorem 1, for α -Hölder continuous reward function with parameters $\{\lambda_\alpha, \alpha\}$, and the $\mathbb{D}_{\sqrt{J}}$ distance, we can bound the immediate reward value error caused due to the use of wildfire

$$err_r \leq (4 \cdot \ln 2)^{\frac{\alpha}{2}} \cdot \lambda_\alpha \cdot \varepsilon_{wf}^\alpha. \quad (6.37)$$

By selecting to re-use a belief with information gap no larger than ε_{wf} , we sacrifice some of the statistical accuracy of our estimate in favor of a substantial save in computation time. As explained in Section 6.1 and illustrated by Figure 6.1.1, an information gap might occur due to inaccurate predictions for lookahead steps $k+1 : k+l$, unrepresentative measurements for lookahead steps $k+l+1 : k+L$, or some combination of them. Beliefs from planning at time k that are being re-used under the wildfire condition, are not being updated to match the information of planning at time $k+l$, or to have representative future measurements for lookahead steps $k+l+1 : k+L$. Instead, they are considered as close enough, thus taken "as is" and the entire process of belief update (Section 6.4.3) is evaded. The wildfire condition is passed from one belief to all of its descendants along the horizon, e.g. if

$b[X_{i|k}]$ has been flagged as meeting the wildfire condition all beliefs originated in $b[X_{i|k}]$ would also be flagged as meeting the wildfire condition without calculating any distances, i.e. the entire subtree rooted at $b[X_{i|k}]$ is taken "as is", without any re-calculations, as discussed below.

It is worth stressing that the wildfire condition is a non-integral part of iX-BSP, and as such, it is up to the user to decide whether to sacrifice statistical accuracy in favor of computation time or not. Moreover it is up to the user to decide how much sacrifice he or she are willing to make by adjusting the wildfire condition accordingly $\varepsilon_{wf} \in [0, \varepsilon_c]$. Choosing a wildfire threshold value of $\varepsilon_{wf} = 0$, does not yield any immediate reward error, but can still save computation time in instances where an identical belief is available from a previous planning session, e.g. BSP under MPC framework with no new available observations from the environment.

6.7.2 USING wildfire WITHIN iX-BSP

We will now meticulously demonstrate how the wildfire condition is integrated within the iX-BSP paradigm. There are two different places within iX-BSP in-which the wildfire condition is used, the first is just after selecting the closest branch for re-use (Alg. 3 line 3), and the second is part of re-using existing beliefs (Alg. 5 line 12).

Let us assume we have just located the closest branch for re-use from planning at time k (Alg. 3 line 1), where Dist is the distance between our last posterior $b[X_{k+l|k+l}]$ and its counterpart from planning time k $\tilde{b}[X_{k+l|k}]$, which is also the root of the selected branch. The value of Dist represents the information gap between the current posterior belief $b[X_{k+l|k+l}]$, and the appropriate closest prediction to it from a precursory planning session, $\tilde{b}[X_{k+l|k}]$. In this specific case, the information gap represents how well were the predictions at planning time k for lookahead steps $k+1 : k+l$, the closer they were to what actually happened, the smaller the information gap as well as Dist value. If the information gap is not too big, i.e. $\text{Dist} \leq \varepsilon_c$ (Alg. 3 line 2), we say that the selected candidate branch is re-use worthy; if the information gap also meets the wildfire condition, i.e. $\text{Dist} \leq \varepsilon_{wf}$ (Alg. 3 line 3), we say that the information gap is negligible, thus the entire selected branch can be re-used "as is". In case of the latter, we re-use the entire closest branch $\mathcal{B}_{k+l|k}$ rooted at $\tilde{b}[X_{k+l|k}]$, and continue to complete the rest of the lookahead steps with X-BSP. Because the beliefs were re-used without any update, there is no need to re-calculate the appropriate immediate reward values, available from planning at time k .

Let us now assume that the information gap was not too big, but also did not meet the wildfire condition, i.e. $\varepsilon_{wf} < \text{Dist} \leq \varepsilon_c$, that scenario takes us to the second use of the wildfire condition in iX-BSP, as part of re-using existing beliefs Alg. 5. Because the wildfire condition is passed from one belief to its descendants, we always start by checking if a candidate belief has inherited a wildfire flag

Variable	Description
$\square_{t k}$	Of time t while current time is k
$b[X_{t k}]$	belief at time t while current time is k
$b^s[X_{t k}]$	The s_{th} sampled belief representing $b[X_{t k}]$
$\tilde{b}[X_{t k}]$	The root of the selected branch for re-use from planning time k
$\mathcal{B}_{t k}$	The set of all beliefs from planning time k rooted in $\tilde{b}[X_{t k}]$
$b_a^{s-}[X_{t+1 k+l}]$	The sampled belief $b^s[X_{t k+l}]$ propagated with the a candidate action
$b_a^{s'-}[X_{t+i k}]$	A propagated belief from $\mathcal{B}_{t k}$ closest to $b_a^{s-}[X_{t+1 k+l}]$
ε_c	belief distance critical threshold, max distance for re-use computational advantage
ε_{wf}	wildfire threshold, max distance to be considered as close-enough for re-use without any update
$\mathbb{D}(\cdot)$	belief divergence / metric
$\mathbb{D}^2(\cdot)$	squared $\mathbb{D}(\cdot)$
$\mathbb{D}_{\sqrt{J}}(p, q)$	The distance between distributions p and q according to the $\mathbb{D}_{\sqrt{J}}$ distance
λ_a	the reward function a – Hölder constant
α	the reward function a – Hölder exponent
$r_{t k}^s$	the immediate reward at lookahead step t , related to $b^s[X_{t k}]$
Dist	The distance between $\tilde{b}[X_{t k}]$ and the corresponding posterior $b[X_{t t}]$
dist	The distance between $b_a^{s'-}[X_{t k}]$ and $b_a^{s-}[X_{t k+l}]$
Δ	equals $\mathbb{D}^2(b_1^+, b_2^+) - \mathbb{D}^2(b_1, b_2)$, where b_{ip} denotes b_i propagated with motion and measurements

Table 6.1: Notations for Section 6.7

from its ancestor (Alg. 5 line 5). In case it did, we automatically consider it as meeting the wildfire condition, and flag its immediate children as such as well. In case a belief is not already flagged as meeting the wildfire condition we are required to check it. We consider the s_{th} belief at lookahead step $t = i$, i.e. $b^s[X_{i|k+l}]$, and propagate it with action a to obtain $b_a^{s-}[X_{i+1|k+l}]$ (Alg. 5 line 9). We locate the closest propagated belief to $b_a^{s-}[X_{i+1|k+l}]$, denote it as $b_a^{s'-}[X_{i+1|k}]$ and the distance between them as dist. In this case, the information gap represented by dist, consists of the gap represented by Dist as well as the possibly different predicted measurements for lookahead steps $(k + l + 1 : i)$, e.g. area (iii) in Figure 6.1.1. Because the propagated belief is used to generate predicted measurements (see Alg. 2), a small enough value of dist would improve the chances to obtain a representative set

of samples (as discussed in Section 6.4.2).

If the information gap is not too big, i.e. $\text{dist} \leq \varepsilon_c$ (Alg. 5 line 11), we consider the previously sampled measurements associated to $b_a^{s'}[X_{i+1|k}]$ as candidates for a representative set of measurements (Alg. 5 lines 17-18). If the information gap also meets the wildfire condition, i.e. $\text{dist} \leq \varepsilon_{wf}$, we consider $b_a^s[X_{i+1|k+l}]$ and $b_a^{s'}[X_{i+1|k}]$ as close enough such that all sampled measurements associated with the latter are representative of the former (Alg. 5 line 13). Consequently, we consider all beliefs descendant of $b_a^{s'}[X_{i+1|k}]$ as meeting the wildfire condition, and as such they are re-used "as is" without any update.

6.7.3 OBJECTIVE VALUE BOUNDS UNDER wildfire

Under the use of wildfire, iX-BSP is not necessarily an exact solution of the BSP problem, but a possible approximation. As such, we would like to get a bound over the resulting objective value. In the following we show that under the assumption of α -Hölder rewards and the use of $\mathbb{D}_{\sqrt{J}}$ distance (Appendix B) the immediate reward value under wildfire is bounded (Theorem 1) by a random variable. We continue with showing that the corresponding objective value is also bounded (Theorem 2) by a (different) random variable, where if the distribution of this random variable is explicitly given, a corresponding bound can be formulated (Theorem 3). We conclude with showing (Corollary 3) that for the case of linear Gaussian models, one can explicitly calculate the moments of the aforementioned random variables. For the reader's convenience Figure 6.7.1 illustrates the workflow of the supplied proofs as well as the dependency of each segment over the two involved assumptions, while bolding the final results - Theorem 2, Theorem 3, and Corollary 3. It is worth reiterating that the purpose of the supplied bounds is to reflect the direct correlation between ε_{wf} and the objective value. More work is required in order to make the bounds convenient enough for online usage, e.g. dynamically updating the ε_{wf} value - we leave this for future work.

Under the assumption of general α -Hölder rewards $r(b, u)$, we can get a bound over the difference between two immediate reward functions of the same action and different beliefs, as stated in Theorem 1.

Corollary 1 (of Theorem 1). *Let $r(b, u)$ be α -Hölder continuous with λ_α and $\alpha \in (0, 1]$. Let b and b' denote two future beliefs. Then the bounded difference between $r(b, u)$ and $r(b', u)$ is a random variable.*

Proof. Using Theorem 1, the bound is given by

$$|r(b, u) - r(b', u)| \leq (4 \cdot \ln 2)^{\frac{\alpha}{2}} \cdot \lambda_\alpha \cdot \mathbb{D}_{\sqrt{J}}^\alpha(b, b'), \quad (6.38)$$

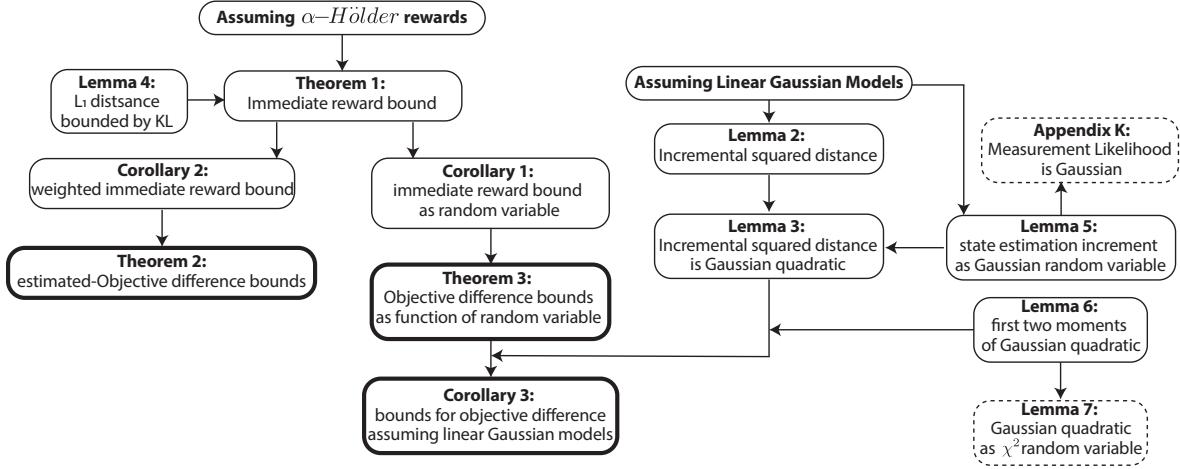


Figure 6.7.1: Illustration of the layout for finding the objective value bounds under wildfire. The bolded rectangles denote three variations for bounds over the objective value under wildfire. Theorem 2 provides the objective bound when calculated using samples rather than expectation, whereas Theorem 3 provides the bound when explicitly solving the expectation (assuming the probability is known). Corollary 3 demonstrate how to explicitly calculate the bound provided in Theorem 3 assuming linear Gaussian models. The dotted rectangles denote additional non-integral insights, deduced along the way.

The $\mathbb{D}_{\sqrt{J}}$ distance is a function of future beliefs b and b' . The future beliefs b and b' are a function of future measurements. Future measurements are a random variable distributed according to the appropriate measurement likelihood $\mathbb{P}(z|H^-)$. As a function of random variables, $\mathbb{D}_{\sqrt{J}}$ is also a random variable. \square

Corollary 2 (of Theorem 1). *Let $r(b, u)$ be α -Hölder continuous with λ_α and $\alpha \in (0, 1]$. Let b and b' denote two beliefs. Let ω_i denote a positive weight, such that $0 \leq \omega_i \leq 1$, $i \in \{1, 2\}$. Then the weighted difference between $r(b, u)$ and $r(b', u)$ is given by*

$$(\omega_1 - \omega_2)r(b') - \omega_1 \bar{\lambda}_\alpha \mathbb{D}_{\sqrt{J}}^{\alpha}(b, b') \leq \omega_1 r(b) - \omega_2 r(b') \leq \omega_1 \bar{\lambda}_\alpha \mathbb{D}_{\sqrt{J}}^{\alpha}(b, b') + (\omega_1 - \omega_2)r(b') \quad (6.39)$$

where

$$\bar{\lambda}_\alpha \triangleq \lambda_\alpha (4 \cdot \ln 2)^{\frac{\alpha}{2}}. \quad (6.40)$$

Proof. see Appendix G. \square

As the objective function is defined by the sum of expected rewards along the planning horizon, we are now in a position to provide a bound over the difference between two objective values considering the same action sequence yet different beliefs.

Theorem 2. *Let $r(b, u)$ be α -Hölder continuous with λ_α and $\alpha \in (0, 1]$. Let $J_{k+l|k+l}$ and $J_{k+l|k}$ be objective values of the same time step $k + l$, calculated based on information up to time $k + l$ and k respectively. Let L be a planning horizon such that $L \geq l + 1$. Let n_i be the number of samples used to estimate the expected*

reward at lookahead step i . Let ω_i^j be non-negative weights such that $0 \leq \omega_i^j \leq 1$ and $\sum_{j=1}^{n_i} \omega_i^j = 1$. Then the difference $(J_{k+l|k+l} - J_{k+l|k})$ is bounded by

$$\sum_{i=k+l+1}^{k+L} \sum_{j=1}^{n_i} \omega_i^j [r_{i|k}^j - \mathcal{D}_i^j] - J_{k+l|k} \leq J_{k+l|k+l} - J_{k+l|k} \leq \sum_{i=k+l+1}^{k+L} \sum_{j=1}^{n_i} \omega_i^j [r_{i|k}^j + \mathcal{D}_i^j] - J_{k+l|k}. \quad (6.41)$$

where

$$\mathcal{D}_i^j = \lambda_a (4 \cdot \ln 2)^{\frac{a}{2}} \mathbb{D}_{\sqrt{J}}^a (b^j[X_{i|k+l}], b^j[X_{i|k}]). \quad (6.42)$$

Proof. see Appendix H. □

In order to calculate the bound, we need to sample a set of future measurements, and any specific realization of such a set may result with a different bound value altogether. In case the probability function of the bound in Corollary 1 can be obtained, there is no call for using samples as in Theorem 2, and the bound over the objective value can be analytically calculated as suggested by Theorem 3. In other words, while Theorem 2 offers a sample based estimation for the objective error bound, Theorem 3 offers the un-approximated formulation. For $n_i \rightarrow \infty$, both theorems will provide exactly the same bound.

Theorem 3. Let $r(b, u)$ be a -Hölder continuous with λ_a and $a \in (0, 1]$. Let $J_{k+l|k+l}$ and $J_{k+l|k}$ be objective values of the same time step $k + l$, calculated based on information up to time $k + l$ and k respectively. Let L be a planning horizon such that $L \geq l + 1$. Then the difference $(J_{k+l|k+l} - J_{k+l|k})$ is bounded by

$$\varphi - \psi \leq J_{k+l|k+l} - J_{k+l|k} \leq \varphi + \psi, \quad (6.43)$$

where

$$\varphi \triangleq \sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} (\omega - 1) r_{i|k} \quad (6.44)$$

$$\psi \triangleq \lambda_a (4 \cdot \ln 2)^{\frac{a}{2}} \left[(L - l) \varepsilon_{wf}^a + \sum_{i=k+l+1}^{k+L} \left(\sum_{j=k+l+1}^i \mathbb{E}_{z \sim p_{k+l}} \Delta_j \right)^{\frac{a}{2}} \right], \quad (6.45)$$

and

$$\mathbb{D}_{\sqrt{J}}(b[X_{i|k+l}], b[X_{i|k}]) = \sqrt{\mathbb{D}_{\sqrt{J}}^2(b[X_{i-1|k+l}], b[X_{i-1|k}]) + \Delta_i}, \quad (6.46)$$

$$\varepsilon_{wf} = \mathbb{D}_{\sqrt{J}}(b[X_{k+l|k+l}], b[X_{k+l|k}]), \quad (6.47)$$

$$\omega = \frac{\mathbb{P}(z_{k+l+1:k+L}|H_{k+l|k+l}, u_{k+l:k+L-1})}{\mathbb{P}(z_{k+l+1:k+L}|H_{k+l|k}, u_{k+l:k+L-1})} \triangleq \frac{p_{k+l}}{p_k}. \quad (6.48)$$

Proof. see Appendix I. □

We can see that in the bound suggested by Theorem 3 even for $\varepsilon_{wf} = 0$ we are still left with a stochastic expression, depending on the measurement likelihood probabilities, whereas for identical measurement likelihoods, i.e. $\omega = 1$, as well as identical beliefs, we expect zero difference between the objective values.

ASSUMING LINEAR GAUSSIAN MODELS

As discussed earlier Δ is a function of the future measurements and as such it is a random variable. In the following we explicitly calculate the first moment of Δ required for calculating the bound of Theorem 3, under the simplifying assumption of linear Gaussian models

$$x' = \mathcal{F}x + \mathcal{J}u + w \quad , \quad w \sim \mathcal{N}(0, \Sigma_w), \quad (6.49)$$

$$z = \mathcal{H}x + v \quad , \quad v \sim \mathcal{N}(0, \Sigma_v), \quad (6.50)$$

where \mathcal{F} and \mathcal{J} are the motion model jacobian in respect to the state and action appropriately, \mathcal{H} is the measurement jacobian, and w and v are zero mean additive gaussian noises.

Lemma 2 (Incremental $\mathbb{D}_{\sqrt{J}}$ distance). *Let b_1 and b_2 be two Gaussian beliefs $\mathcal{N}(\mu_1, \Sigma_1)$ and $\mathcal{N}(\mu_2, \Sigma_2)$, respectively with state dimension d , and their two (differently) propagated counterparts b_{1p} and b_{2p} with $\mathcal{N}(\mu_{1p}, \Sigma_{1p})$ $\mathcal{N}(\mu_{2p}, \Sigma_{2p})$ and with state dimension d_p . There exist ζ_i and A_i such that the propagated mean and covariance are given by*

$$\mu_{ip} = \mu_i + \zeta_i \quad , \quad \Sigma_{ip} = (\Sigma_i^{-1} + A_i^T A_i)^{-1} \quad , \quad i \in [1, 2]. \quad (6.51)$$

Then the squared $\mathbb{D}_{\sqrt{J}}$ distance between the propagated beliefs can be written as

$$\mathbb{D}_{\sqrt{J}}^2(b_{1p}, b_{2p}) = \mathbb{D}_{\sqrt{J}}^2(b_1, b_2) + \Delta, \quad (6.52)$$

where

$$\begin{aligned}\Delta = & \frac{1}{4}(\mu_2 - \mu_1)^T [A_2^T A_2 + A_1^T A_1] (\mu_2 - \mu_1) + \frac{1}{2}(\mu_2 - \mu_1)^T [\Sigma_{1p}^{-1} + \Sigma_{2p}^{-1}] (\zeta_2 - \zeta_1) \\ & + \frac{1}{4}(\zeta_2 - \zeta_1)^T [\Sigma_{1p}^{-1} + \Sigma_{2p}^{-1}] (\zeta_2 - \zeta_1) + \frac{1}{4} \text{tr} (A_2^T A_2 \Sigma_{1p} - \Sigma_2^{-1} \Sigma_1 A_1^T (I + A_1 \Sigma_1 A_1^T)^{-1} A_1 \Sigma_1) \\ & + \frac{1}{4} \text{tr} (A_1^T A_1 \Sigma_{2p} - \Sigma_1^{-1} \Sigma_2 A_2^T (I + A_2 \Sigma_2 A_2^T)^{-1} A_2 \Sigma_2) - \frac{1}{2} (d_p - d). \quad (6.53)\end{aligned}$$

Proof. see Appendix J. \square

Next we make the observation that Eq. (6.53) is a quadratic form of a Multivariate Gaussian vector,

Lemma 3 (Incremental $\mathbb{D}_{\sqrt{J}}$ distance as Gaussian Quadratic). *Let b_1 and b_2 be two Gaussian beliefs $\mathcal{N}(\mu_1, \Sigma_1)$ and $\mathcal{N}(\mu_2, \Sigma_2)$, respectively with state dimension d , and their two propagated counterparts b_{1p} and b_{2p} with $\mathcal{N}(\mu_{1p}, \Sigma_{1p})$ $\mathcal{N}(\mu_{2p}, \Sigma_{2p})$ and with state dimension d_p . There exist ζ_i and A_i such that the propagated mean and covariance are given by,*

$$\mu_{ip} = \mu_i + \zeta_i \quad , \quad \Sigma_{ip} = (\Sigma_i^{-1} + A_i^T A_i)^{-1} \quad , \quad i \in [1, 2]. \quad (6.54)$$

Then the incremental $\mathbb{D}_{\sqrt{J}}$ distance $\Delta \triangleq \mathbb{D}_{\sqrt{J}}^2(b_{1p}, b_{2p}) - \mathbb{D}_{\sqrt{J}}^2(b_1, b_2)$ is a quadratic form of a gaussian variable.

Proof. see Appendix K. \square

We are now in position to explicitly calculate the bounds of Theorem 3 under the assumption of linear Gaussian models,

Corollary 3 (of Theorem 3). *Let $r(b, u)$ be α -Hölder continuous with λ_α and $\alpha \in (0, 1]$. Let $J_{k+l|k+l}$ and $J_{k+l|k}$ be objective values of the same time step $k+l$, calculated based on information up to time $k+l$ and k respectively. Let L be a planning horizon such that $L \geq l+1$. Let the motion and measurement models be linear with additive Gaussian noise (6.49)-(6.50). Then the bound of $(J_{k+l|k+l} - J_{k+l|k})$ can be explicitly calculated.*

Proof. see Appendix M. \square

6.8 RESULTS - WILDFIRE

In this section we examine the effects of wildfire over the iX-BSP paradigm. We compare iX-BSP with and without wildfire (see Section 6.8.1) in the sense of planning-session computation time, the

Prior belief standard deviation	$\begin{bmatrix} 1^\circ \cdot I_{3 \times 3} & \text{o} \\ \text{o} & 5[m] \cdot I_{3 \times 3} \end{bmatrix}$
Motion Model standard deviation	$\begin{bmatrix} \text{o}.5^\circ \cdot I_{3 \times 3} & \text{o} \\ \text{o} & \text{o}.5[m] \cdot I_{3 \times 3} \end{bmatrix}$
Observation Model standard deviation	$\begin{bmatrix} 3[\text{px}] & \text{o} \\ \text{o} & 3[\text{px}] \end{bmatrix}$
Camera Aperture	90°
Camera acceptable Sensing Range	between $2[m]$ and $4o[m]$
useWF	true
ε_c	250
ε_{wf}	2
β_σ	1.5
n_u	3
n_x	5
n_z	1
action primitives	left, right and forward with $1[m]$ translation and $\pm 90^\circ$ rotations
\mathbb{D}	$\mathbb{D}_{\sqrt{J}}$

Table 6.1: Parameters for Section 6.8.1 following Alg. 3

posterior estimation error upon reaching the goal and the covariance norm upon reaching the goal. We do so on the exact scenario used in Section 6.6.2, with the single exception of using wildfire. We also perform a sensitivity analysis for the wildfire threshold value ε_{wf} (see Section 6.8.2), in order to check its effect over the objective value.

6.8.1 wildfire EFFECT ON PERFORMANCE

We compare iX-BSP with and without the use of wildfire under the same scenario as in Section 6.6.2 (see parameters in Table 6.1). We use the same code and the same Linux machine to perform 20 rollouts, each with a different sampled ground-truth for the prior state, on the same 10 maps presented in Figure 6.6.3a, and with the same reward function (6.28). Figure 6.8.1a presents a box-plot of the accumulated planning time of all 200 rollouts (20 rollouts over each of the 10 maps), where the computation time advantage in favor of wildfire usage is easily noticed, on average wildfire saved 90% off iX-BSP accumulated planning time. While iX-BSP favors in computation time over X-BSP due

to a more efficient belief update, the significant reduction in computation time when using wildfire is originated in the fact that for "close enough" beliefs we refrain from updating the belief altogether. As previously mentioned, the use of wildfire with a non-zero threshold is an approximation of iX-BSP and essentially of X-BSP. As such, we would expect that using wildfire will affect the estimation accuracy and covariance, but as seen in Figures 6.8.1b-6.8.1c, there is no significant toll on estimation. The reason for this supposedly "free lunch" hides behind the choice of the wildfire threshold, as the objective value error due to the use of wildfire is directly related to the choice of wildfire threshold (as seen in Section 6.7.3). For a small enough wildfire threshold the same action is chosen, and so the impact over the estimation is practically unnoticeable. It is worth stressing out that this will not always be the case, as wildfire is in-fact an approximation, and one should treat it as such when choosing to invoke it. Moreover wildfire can be seen as breaking the MPC framework, whenever the newly obtained information of time k is "close enough" to the most relevant prediction of time k we use the prediction rather than updating it with the new information. We continue with empirically testing the impact of the wildfire threshold over the objective value in Section 6.8.2.

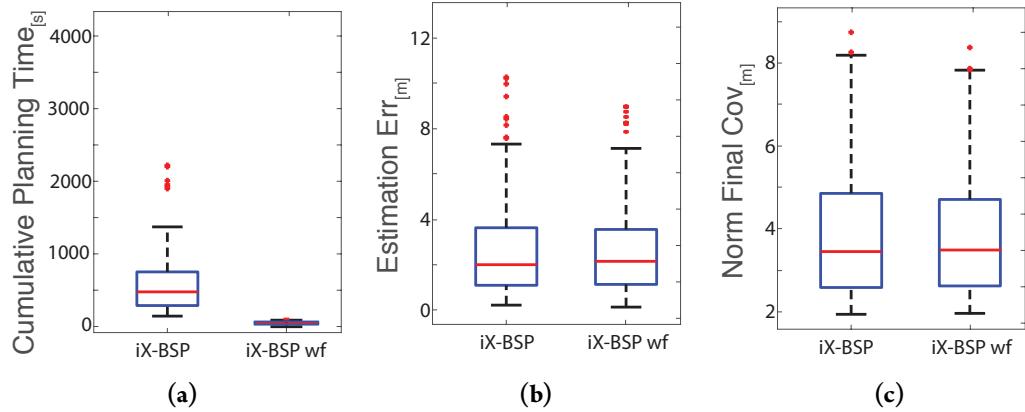


Figure 6.8.1: Statistical comparison between iX-BSP with and without wildfire over the same scenario as in Figure 6.6.3 of 10 randomly generated maps each with 2 goals and between 2 to 150 landmarks. (a) (b) and (c) Box plots of 200 rollouts for planning session timing (a), posterior estimation error upon reaching the goal (b) and the covariance norm upon reaching the goal (c).

6.8.2 wildfire THRESHOLD - SENSITIVITY ANALYSIS

Complementary to the bounds provided in Section 6.7.3, we provide an empiric analysis for the impact the wildfire threshold holds over the objective value, for the non-Lipschitz reward function (6.28). Code implemented in MATLAB using iSAM2 efficient methodologies and executed on the same Linux machine. The relevant parameters are summarized in Table 6.2.

Prior belief standard deviation	$\begin{bmatrix} 1^\circ \cdot I_{3 \times 3} & \text{o} \\ \text{o} & 5[m] \cdot I_{3 \times 3} \end{bmatrix}$
Motion Model standard deviation	$\begin{bmatrix} \text{o}.5^\circ \cdot I_{3 \times 3} & \text{o} \\ \text{o} & \text{o}.5[m] \cdot I_{3 \times 3} \end{bmatrix}$
Observation Model standard deviation	$\begin{bmatrix} 3[px] & \text{o} \\ \text{o} & 3[px] \end{bmatrix}$
Camera Aperture	90°
Camera acceptable Sensing Range	between $z_{[m]}$ and $40_{[m]}$
n_u	3
n_x	5
n_z	1
action primitives	left, right and forward with $1_{[m]}$ translation and $\pm 90^\circ$ rotations

Table 6.2: Parameters for Section 6.8.2 following X-BSP

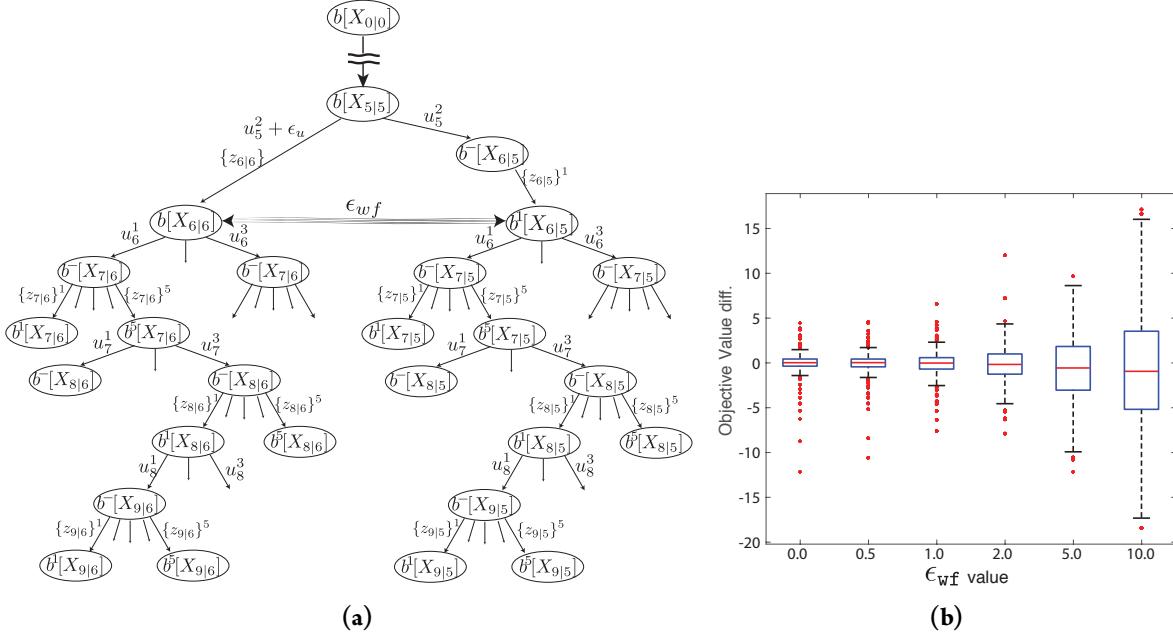


Figure 6.8.2: Statistical evaluation of wildfire threshold effect over Objective value. (a) The belief trees used for the statistical evaluation: the shared history up to time $t = 5$; the planning tree at time $t = 5$ on the right; the planning tree at time $t = 6$ on the left. (b) Box plot of the objective value difference between corresponding action sequences as a function of the forced distance between $b[X_6|6]$ and $b[X_6^3|5]$.

In order to perform such analysis, we would like to compare the objective values resulting from planning over two posterior beliefs, while the $\mathbb{D}_{\sqrt{f}}$ distance between them equals different wildfire

threshold values. The scenario used for this analysis is illustrated in Figure 6.8.2a. For six different values of $\varepsilon_{wf} = \{0, 0.5, 1, 2, 5, 10\}$, and each of the 10 maps in Figure 6.6.3a we perform 20 repetitions of the following. Starting from the same prior belief $b[X_{0|0}]$, and the same ground-truth, we advance the robot 5 steps forward and obtain the belief $b[X_{5|5}]$. Using $b[X_{5|5}]$ we obtain two different beliefs, the first by propagating $b[X_{5|5}]$ with the primitive action u^2 and the resulting *sampled* measurements $\{z_{6|5}\}^1$ - denoted as $b^1[X_{6|5}]$, and the second by calculating the posterior $b[X_{6|6}]$ resulting from advancing the robot using the same primitive action u^2 along with some additive noise ε_u and obtaining the corresponding measurements $\{z_{6|6}\}$. We now have two beliefs, $b^1[X_{6|5}]$ and $b[X_{6|6}]$, both representing the belief at time $t = 6$ but with partially different history, the same we would get from performing iX-BSP under MPC framework. The additive noise ε_u is chosen such that it would cause the $\mathbb{D}_{\sqrt{J}}$ distance between $b^1[X_{6|5}]$ and $b[X_{6|6}]$ to equal the desired wildfire threshold value - ε_{wf} . We then perform X-BSP over both $b^1[X_{6|5}]$ and $b[X_{6|6}]$, with the same parameters, and obtain corresponding objective values for each of the candidate action sequences. As both planning trees share the same action sequences we can compare the corresponding objective values per candidate action sequence. We denote the difference between the aforementioned as the Objective Value difference, which form the vertical axes in Figure 6.8.2b.

Figure 6.8.2b presents the results of the described analysis, the objective value difference as a function of the $\mathbb{D}_{\sqrt{J}}$ distance between $b^1[X_{6|5}]$ and $b[X_{6|6}]$ denoted as ε_{wf} . We can clearly see the variance of the different objective value differences increasing with the wildfire threshold values, thus reflecting a connection between the objective value difference and the wildfire threshold value over non-Lipschitz reward. Moreover, for the results in Figure 6.8.2b, the relation between the objective value difference variance and the wildfire threshold values appears to be somewhat linear, but more work is needed before anything can be concretely deduced.

6.9 SOME BROADER PERSPECTIVE

In this section we briefly discuss the motivation for iX-BSP and provide some broader perspective to possible future usage. As mentioned earlier, the iX-BSP paradigm deals with incrementally calculating the current belief space planning session through the selective re-use of sampled measurements from previous planning sessions. By re-using sampled measurements, we can re-use the appropriate future beliefs already calculated in said previous planning session and update them with current information, thus averting from the time consuming standard inference over these future beliefs.

In our work we considered assumptions 1-3, and based on them validated our theoretical work empirically while comparing the performance of iX-BSP to the extent of the horizon overlap.

As discussed earlier, decision making under uncertainty in high dimensional state space is com-

putationally intractable, and as such the majority of the plan-act-infer computation time can be ascribed to it. While saving computation time without effecting the solution accuracy, as well as having a controlled mechanism to sacrifice accuracy in favor of performance is of great importance, the aforementioned assumptions might paint a picture of a case sensitive formulation in which iX-BSP can function, is it truly so? Moreover, when considering the entire planning session computation time, i.e. beyond the extent of the horizon overlap, the advantage of iX-BSP without wildfire, i.e. unapproximated formulation, over X-BSP might feel somewhat marginal, can it be further improved without resorting to approximations?

In the following we answer those questions starting with discussing the source of assumptions 1-3.

In its general form, iX-BSP requires a set of candidate beliefs for re-use (Alg. 3 line 1), and a method of choosing the closest belief from this set (Section 6.3.2). In this work we use a previous planning session as a candidate set (assumption 1) and use assumptions 2-3 to further prune it into a smaller candidate set. Removing assumptions 2-3 would entail the use of an efficient method to store and search the set of candidate beliefs for the closest one. So while access to a candidate set for re-use (assumption 1), whether it is some previous planning session or a tailor-made set of candidate beliefs, is an integral part of iX-BSP, the rest of the assumptions can in-fact be omitted.

Secondly, the performance of iX-BSP can be traced back to the validity of the set of belief candidates for re-use. On the one hand, if the set is too far off iX-BSP would perform like X-BSP and on the other hand if we can always find candidate beliefs that are identical to the current posterior the computation time of iX-BSP would be reduced to a bare minimum - locating the candidate belief. Under the "basic" scenario we chose to address, where the candidate set is of previous planning session, the map is previously unknown, there is no available mechanism for predicting new states and the use of a simple naive mechanism for predicting existing states, the performance of iX-BSP is bounded by the number of factors the planning process could properly predict. For example, let us assume that 50% of the factors from last inference are related to new states, it means that even in the best case scenario, where previous planning session correctly predicted all existing states, iX-BSP would still need to update 50% of posterior factors for each future belief along the planning tree. Moreover, the use of a naive mechanism for predicting existing states can (and in-fact do) cause a "bad" prediction with factors related to existing states needed to be updated or even removed, resulting in increasing the computation time per future belief along the planning tree.

In order to further improve the computational advantage of iX-BSP without introducing any approximations we would need to address the computational load of locating and updating candidate beliefs. Using a more efficient way to store and search the candidate set of beliefs could reduce a considerable amount of computation time as the search is being done for each of the future beliefs along the planning horizon. Regarding the incremental update of future beliefs, the naive approach

to reduce computational load would be to use "closer" beliefs as candidates. Either by improving the prediction mechanisms of the planning process for both existing and new states, or by considering a bigger set of candidate beliefs or even tailor-made to allow a discretization of the belief space in a way that we would always have a belief that is close up to some pre-determined ε to the current posterior. The latter would require an extremely efficient paradigm for storing and searching the set of candidate beliefs.

To sum up, other than requiring some set of candidate beliefs, the aforementioned assumptions can in-fact be omitted. Furthermore, the computation time advantage of iX-BSP can be improved without introducing any approximations by improving the handling of the set of candidate beliefs, and reducing its distance from the current posterior. It is worth reiterating that iX-BSP is not limited to using previous planning session as candidate for re-use, although it is readily available on a plan-act-infer system and relieve the user from dealing with concocting a tailor-made set of candidates.

6.10 CONCLUDING iX-BSP

State of the art approaches under X-BSP paradigm (BSP with expectation) calculate each planning session from scratch. In this chapter we suggested to re-formulate the general problem of BSP using expectation, X-BSP. We presented iX-BSP, incrementally calculating the expectation by utilizing previous planning sessions in order to solve the current planning session with a reduced computational effort and without affecting the solution accuracy.

By selectively re-using sampled measurements from previous planning sessions, we are able to avert from standard Bayesian inference as part of reward(cost) values thus reducing the computational effort. As the expectation in iX-BSP is potentially considered over a set of samples taken from multiple different measurement likelihood distributions, we reformulate X-BSP as a Multiple Importance Sampling (MIS) problem, thus statistically maintaining the solution accuracy. Considering their stochastic nature, we evaluate iX-BSP against X-BSP in simulation considering active-SLAM as application, comparing both cumulative planning computation time and estimation accuracy upon reaching the goal. By considering different sampled ground-truth prior states we are able to show that iX-BSP is statistically equal to X-BSP whilst providing shorter computation time.

In addition to providing with the full formulation of iX-BSP, we introduce a non-integral approximation denoted as wildfire, enabling one to trade accuracy for computation time by putting a threshold defining beliefs as "close enough" to be considered as identical. We also analyze, analytically and empirically, the effect wildfire holds over the resulting objective value, as well as demonstrate using wildfire in iX-BSP. Because iX-BSP changes the solution approach of the original, un-approximated, problem (X-BSP), we believe it can be utilized to also reduce computation time

of existing approximations of X-BSP. To support this claim we push further and show how it can be utilized to benefit an existing common approximation of X-BSP. In Chapter 7 we present iML-BSP which provides reduced computation time compared to ML-BSP, while obtaining the same estimation accuracy. We demonstrate the performance advantage iML-BSP holds over ML-BSP in both simulation and real-world experiments considering vision-based active-SLAM in previously unknown uncertain environment and high dimensional state space.

In contrast to common research directions dealing with approximations of the X-BSP problem, we tackled the un-approximated formulation of X-BSP and suggested to improve it by considering calculation re-use across planning sessions, thus enabling to reduce X-BSP computation time without affecting accuracy. As iX-BSP is equivalent to X-BSP, we claim that existing approximations of X-BSP could benefit from the iX-BSP formulation, as demonstrated by considering the ML approximation ML-BSP and the resulting iML-BSP. While demonstrated here only on data from a precursory planning session, using the same formulation it can be easily shown that iX-BSP can selectively re-use any supplied data, from a set of offline calculated beliefs to planning sessions of other agents, all while maintaining the same estimation accuracy as well as the computational advantage. The performance of iX-BSP can be further improved without introducing any approximations, e.g. by using a more sophisticated prediction mechanism to reduce the number of removed factors, or by using a mechanism to predict factors related to new states.

Expert:

a man who makes three correct guesses consecutively.

Laurence J. Peter

7

iML-BSP

SEEING THAT OUR NOVEL iX-BSP APPROACH CHANGES THE SOLUTION PARADIGM FOR THE ORIGINAL, UN-APPROXIMATED, PROBLEM (X-BSP), WE CLAIM IT COULD BE UTILIZED TO ALSO REDUCE COMPUTATION TIME OF EXISTING APPROXIMATIONS OF X-BSP. To support our claim, in this chapter we present the implementation of iX-BSP principles over a commonly used approximation, ML-BSP, and denote it as iML-BSP. We provide the formulation for the iML-BSP problem (Section 7.1), and provide both statistical analysis over synthetic data (Section 7.2) and live experiments (Section 7.3), comparing iML-BSP to ML-BSP.

7.1 IML-BSP FORMULATION

Under the ML assumption we consider just the most likely measurements, rather than sampling multiple measurements; hence, there is a single importance sampling distribution at each planning step i.e. $M_i = 1 \forall i$, because a single measurement is considered for each action at each time step. Consider-

ering the aforementioned, for the case of iML-BSP, Eq. (6.21) is reduced to

$$J^{iML}(u') \approx \sum_{i=k+l+1}^{k+l+L} [w_i \cdot r_i(b[X_{i|k+l}], u'_{i-1|k+l})], \quad (7.1)$$

and Eq. (6.22), representing the weight at time i is reduced to

$$w_i = \frac{\mathbb{P}(z_{k+l+1:i}|H_{k+l|k+l}, u_{k+l:i-1|k+l})}{q(z_{k+l+1:i})}, \quad (7.2)$$

where $q(\cdot)$ is the importance sampling distribution related to the ML measurement set $z_{k+l+1:i}$, and $\mathbb{P}(z|H, u)$ is the nominal distribution from which the ML measurement set $z_{k+l+1:i}$ should have been taken from. When we take the ML measurements from the nominal distribution, i.e. as in ML-BSP, $w_i = 1, \forall i$, and Eq. (7.1) is identical to Eq. (5.16). More specifically without loosing generality, when considering the planning tree from planning at time k as candidate for re-use, the possible values of the importance sampling distribution are

$$q(z_{k+l+1:i}) = \begin{cases} \mathbb{P}(z_{k+l+1:i}|H_{k+l|k}, u_{k+l:i-1|k}) & \mathbb{D}(b^-[X_{i|k}], b^-[X_{i|k+l}]) \leq \varepsilon_c \\ \mathbb{P}(z_{k+l+1:i}|H_{k+l|k+l}, u_{k+l:i-1|k+l}) & \varepsilon_c < \mathbb{D}(b^-[X_{i|k}], b^-[X_{i|k+l}]) \end{cases}, \quad (7.3)$$

where $b[X_{i|k}]$ is the belief from planning at time k considered for re-use, $\mathbb{P}(z_{k+l+1:i}|H_{k+l|k}, u_{k+l:i-1|k})$ is the measurement likelihood probability used to sample future measurements considered in $b[X_{i|k}]$, $\mathbb{P}(z_{k+l+1:i}|H_{k+l|k+l}, u_{k+l:i-1|k+l})$ is the nominal measurement likelihood probability used to sample future measurements considered in $b[X_{i|k+l}]$, ε_c is the re-use threshold (see Section 6.3.3), and $\mathbb{D}(\cdot)$ denote some belief distance.

By considering an iML-BSP session as a single rollout, it can be extended to rollout-based planners with a belief dependent reward in a straightforward manner, we leave this for future work.

The reader can use the same algorithm provided for iX-BSP (see Alg. 3) with parameters appropriate to iML-BSP, e.g. $n_x = n_z = 1$.

7.2 SIMULATION RESULTS - iML-BSP

In this section we compare ML-BSP and iML-BSP (see parameters in Table 7.1) in the sense of planning-session computation time, the posterior estimation error upon reaching the goal, and the covariance norm upon reaching the goal. For comparison we performed 1000 rollouts (entire mission run), each with a different sampled ground-truth for the prior state. Figure 7.2.1a presents the

scenario on which all rollouts were performed, along with the estimation results of an arbitrary rollout.

Prior belief standard deviation	$\begin{bmatrix} 1^\circ \cdot I_{3 \times 3} & \text{o} \\ \text{o} & S[m] \cdot I_{3 \times 3} \end{bmatrix}$
Motion Model standard deviation	$\begin{bmatrix} \text{o}.S^\theta \cdot I_{3 \times 3} & \text{o} \\ \text{o} & \text{o}.S[m] \cdot I_{3 \times 3} \end{bmatrix}$
Observation Model standard deviation	$\begin{bmatrix} 3[\text{px}] & \text{o} \\ \text{o} & 3[\text{px}] \end{bmatrix}$
Camera Aperture	90°
Camera acceptable Sensing Range	between $2[m]$ and $4o[m]$
useWF	false
ε_c	250
β_σ	1.5
n_u	3
n_x	1
n_z	1
action primitives	left, right and forward with $1[m]$ translation and $\pm 90^\circ$ rotations
\mathbb{D}	$\mathbb{D}_{\sqrt{J}}$

Table 7.1: Parameters for Section 7.2 following Alg. 3 under iML-BSP

A robot equipped with a stereo camera, is tasked with reaching two goals, numbered and denoted by blue dots in Figure 7.2.1a, in a world with 45 randomly placed landmarks, denoted by green crosses in Figure 7.2.1a, while considering the reward function (6.28). Same as in Section 6.6.2, both ML-BSP and iML-BSP consider 6 DOF robot pose, 3 DOF landmarks, a joint state comprised of both robot pose and landmarks and three candidate actions for each step (left, right, forward), hence for the considered horizon of 3 look ahead steps, there are 27 candidate action sequences for each planning session. Under the ML assumption, both methods, ML-BSP and iML-BSP, consider a single measurement per action per look ahead step. While for ML-BSP, this action is always the most likely measurement, hence zero innovation, for iML-BSP this measurement is usually not the most likely measurement, hence we retain some innovation along the look ahead steps.

Similarly to X-BSP and iX-BSP, when considering a plan-act-infer system, all differences between ML-BSP and iML-BSP are confined within the planning block, hence the computation time

of the planning process is adequate for fair comparison.

Figure 7.2.1b presents a box-plot for the timing data of all 1000 rollouts, with 3 and 2 outliers for ML-BSP and iML-BSP respectively, where the computation time advantage is in favor of iML-BSP by a factor of 5. The significant reduction in computation time is originated in the fact that iML-BSP performs inference update in a more efficient way, computation wise, compared to ML-BSP. By considering previously calculated beliefs, and utilizing one of which for efficient inference update instead of performing inference from scratch as done in ML-BSP.

Since we claim to provide a more efficient paradigm than ML-BSP, we also verify how iML-BSP favors in estimation results. Figure 6.6.3c presents a box plot of the estimation error upon reaching the goal for each of the methods. The estimation error was calculated using (6.29).

As can be seen in Figure 7.2.1c both methods average around an estimation error of $3.5[m]$, while only ML-BSP is with a single outlier. In 51.0% of the rollouts, iML-BSP provided with a better estimation error than ML-BSP. The large estimation variance that can be seen in Figure 7.2.1c is similar to the one obtained in Figure 6.6.3c, so it provides some assurance regarding the assumption it is the result of using a small number of samples for estimating the objective. Nonetheless, also here, the empiric estimation variance of both methods can be considered as statistically identical for all practical purposes. Of course a more rigorous examination in required, by analytically comparing the estimation variance, we leave this for future work as well. We push further and compare the covariance norm of the final pose. As can be seen in Figure 7.2.1d, they average around $5.0[m]$, without any outliers, and can be clearly considered as statistically identical for all practical purposes.

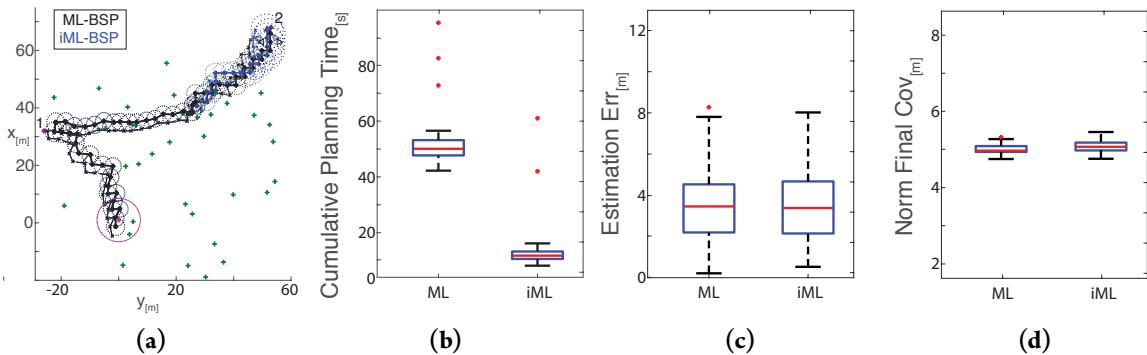


Figure 7.2.1: Statistical comparison between ML-BSP and iML-BSP: (a) The scenario used for the statistical comparison, with 2 numbered goals denoted by blue dots, and 45 landmarks denoted by green crosses. Showcasing the estimation results for one of the 1000 performed rollouts. (b) (c) and (d) Box plots of 1000 rollouts for planning session timing (b), posterior estimation error upon reaching the goal (c) and the covariance norm upon reaching the goal.

Prior belief standard deviation	$\begin{bmatrix} 1^\circ \cdot I_{3x3} & \textcircled{o} \\ \textcircled{o} & S[m] \cdot I_{3x3} \end{bmatrix}$
Motion Model standard deviation	$\begin{bmatrix} 0.5^\circ \cdot I_{3x3} & \textcircled{o} \\ \textcircled{o} & 0.5[m] \cdot I_{3x3} \end{bmatrix}$
Observation Model standard deviation	$\begin{bmatrix} 3[px] & \textcircled{o} \\ \textcircled{o} & 3[px] \end{bmatrix}$
Camera Aperture	90°
Camera acceptable Sensing Range	between $2[m]$ and $40[m]$
useWF	false
ε_c	250
β_σ	1.5
planning horizon	4
n_u	3
n_x	1
n_z	1
action primitives	left, right and forward with $1[m]$ translation and $\pm 45^\circ$ rotations
\mathbb{D}	$\mathbb{D}_{\sqrt{J}}$
ORB-SLAM2	default parameters

Table 7.1: Parameters for Section 7.3 following Alg. 3 under iML-BSP

7.3 REAL WORLD EXPERIMENTS - iML-BSP

In this section we compare ML-BSP and iML-BSP in a real-world setting. In the following we describe the scenario on which we ran these experiments (Section 7.3.1), as well as the results of the two live experiments (Section 7.3.2). All relevant parameters used for these experiments are summarized in Table 7.1.

7.3.1 THE SCENARIO

For these experiment we used the Pioneer 3AT robot, equipped with a ZED stereo camera, Hokuyo UTM-30LX Lidar, and a Linux machine with octa-core i7-6820HQ 2.7GHz processor and 32GB of memory (see Figure 7.3.1).



Figure 7.3.1: Pioneer 3AT robot used for the live experiments. The robot is equipped with a ZED stereo camera, Hokuyo UTM-30LX Lidar, and a Linux machine with octa-core i7-6820HQ 2.7GHz processor and 32GB of memory.

The robot received a number of goals, his mission - reaching each one of the goals while maximizing information gain and minimizing distance to goal using the reward function (6.28). The robot state was comprised of both its own poses and environment landmarks. The robot did not possess any prior information over the environment, nor did it make use of any offline calculations. The robot did receive a prior over its initial pose (as stated in Table 7.1). The environment used for these experiments was the surrounding garden and path-way of a cottage house, laced with both still obstacles (e.g. cars, trees, rocks, children toys, garden furniture) and dynamic obstacles (e.g. people, children, dogs).

The robot uses a plan-act-infer architecture, similar to the one presented in Figure 6.6.1. The planner is the exact same MATLAB code used for the aforementioned simulations where the rest of the code is C++ based. The Hokuyo UTM-30LX Lidar was used for collision avoidance and odometry while the ZED camera was used strictly for its rectified stereo images output. We also made use of ORBSLAM2 as the vision pipeline, feeding the rectified stereo images to ORBSLAM2 and extracting the appropriate factors created by ORBSLAM2 as output.

Future landmark observations are generated by considering only landmarks projected within the camera field of view using posterior estimates for landmark positions and camera pose. As in this work the planning phase considers only the already-mapped landmarks, without reasoning about expected new landmarks, each new landmark observation in inference would essentially mean facing a factor that can not be re-used.

As this is not a simulated environment, where the uncertainties can be replicated, in order to provide a fair comparison between ML-BSP and iML-BSP each planning session the robot performs

both ML-BSP and iML-BSP sequentially, using the same posterior information. The planning duration is timed for comparison, the optimal action given by iML-BSP is being used as the next action, and the optimal action given by ML-BSP is matched against former for comparison.

7.3.2 REAL WORLD RESULTS

In this section we cover the results of live experiments done over two different sets of goals following the scenario described above. Differently than the simulation related results presented so far, we compare the computation time of the entire planning horizon and do not omit the last horizon step. Figure 7.3.2 presents the estimated route of the first experiment, coursing through 4 goals along 35 meters for under both ML-BSP and iML-BSP. As the two methods chose the same optimal action sequences throughout the mission, they have an identical estimation (up to some machine noise). Figure 7.3.3a presents a bar plot of the cumulative planning time for the entire mission for both methods, where each bar is divided into the contribution of the first three horizon steps (denoted in gray) and the contribution of the last horizon step (denoted in black). This division is meant to help the reader compare the live experiment to the simulation results which omitted the shared last horizon step computation time, while still assessing the overall reduction in computation time. The percentage in Figure 7.3.3a represents the relative contribution of the two segments to each of the cumulative planning times. Although the last horizon remains unchanged, we can clearly see the computation time reduction in iML-BSP when compared to ML-BSP from forming 59% of the computation time to only 23%. Figure 7.3.3b presents the planning time per planning session. As expected, in the first planning session considering a new goal iML-BSP performs a regular ML-BSP planning hence both computation times are identical and no factors are re-used.

Figure 7.3.4 suggests some insight on the timing result of iML-BSP by comparing the number of factors involved in the computation of each method. Figure 7.3.4a the sum of added factors per planning session. In blue the number of factors added at time $k+1|k+1$ as part of standard Bayesian inference update. In red the portion of aforementioned factors that are already part of the state prior to the inference update, and as in this work we do not make use of any mechanism to predict new states it also represents the upper bound for the number of factors we can hope to re-use. In orange the number of factors that were originally calculated in a previous planning session and where reused. The difference between the orange and blue lines represents the number of factors needed to be added to the re-used planning tree in order to match the posterior at $k+1$ and the black line in Figure 7.3.4b represents the number of factors needed to be removed from the re-used planning tree. While the orange line is quite close to the red upper bound, there are still a lot of factors needed to be removed in order to match the posterior at $k+1$, which contributes to DA update computation time in iML-BSP.

Using a more sophisticated prediction mechanism for future factors might reduce this overhead in removed factors and save more valuable computation time without introducing an approximation, we leave this for future work.

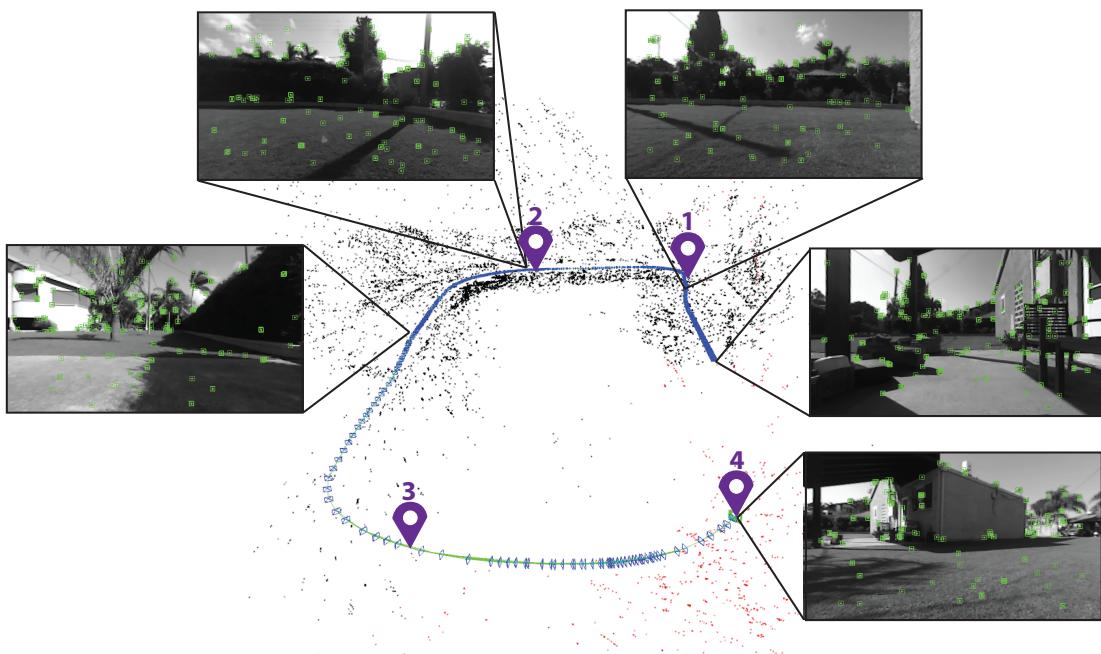


Figure 7.3.2: The first live experiment stretching across a 35 meter course, where the Pioneer robot was given 4 goals (numbered and denoted in purple) to reach. The state estimation of the robot in the form of keyframes (denoted in blue frames), robot trajectory (denoted in green) landmarks (denoted in black and red), and few snapshots along the rout containing the selected features.

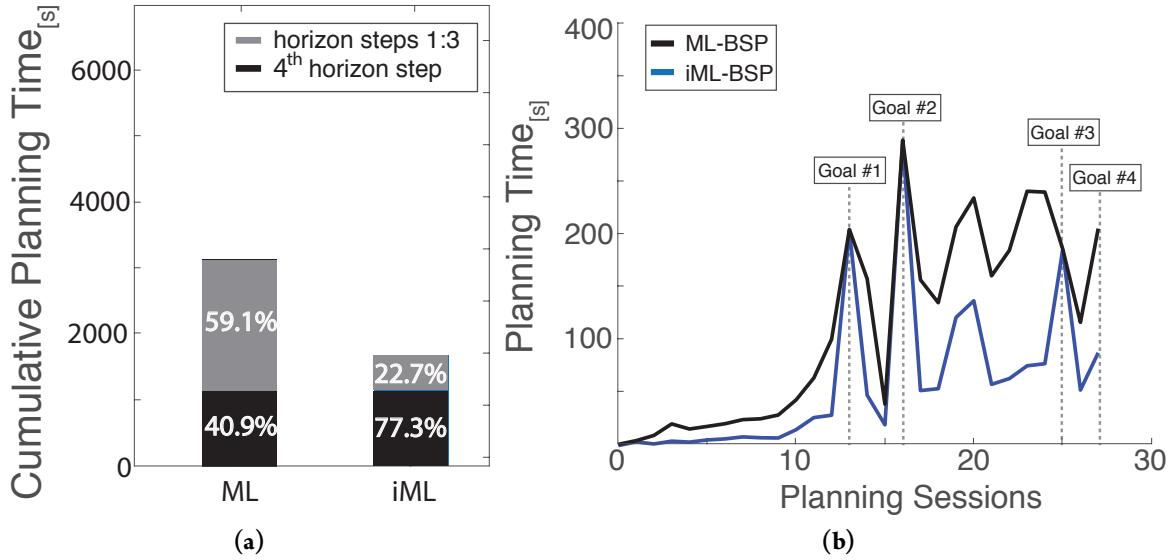


Figure 7.3.3: Planning time results of the first live experiment using the Pioneer robot (a) The cumulative planning time of both ML-BSP and iML-BSP, divided into relative contributions of the first three planning horizon steps (denoted in gray) and the last planning horizon step (denoted in black). (b) Per planning session comparison of the computation time. The goals are marked over the planning session performed after reaching them.

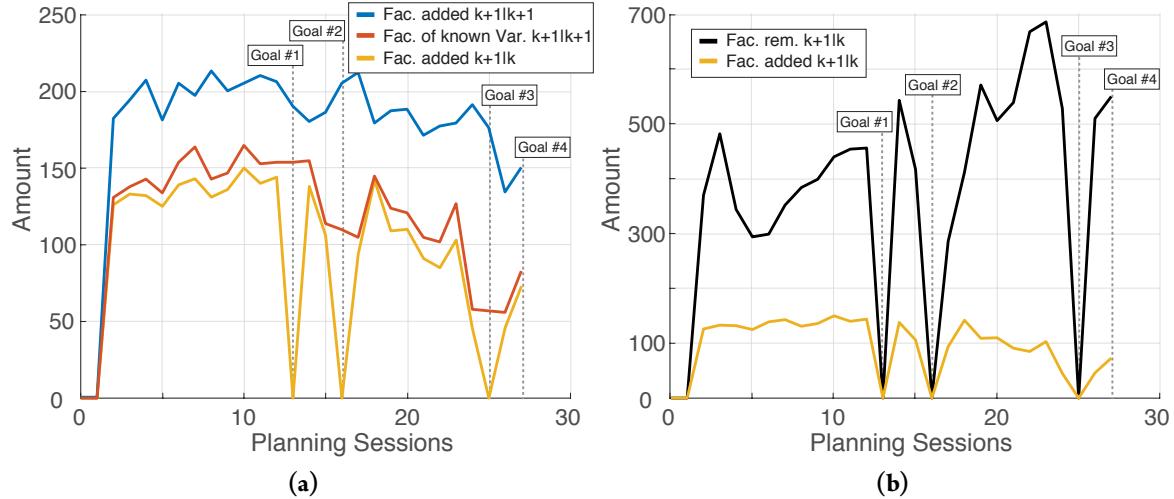


Figure 7.3.4: Number of factors involved in each planning session of the first live experiment. (a) The number of factors added in the last inference update at $k + 1|k + 1$, denoted in blue. The portion of aforementioned factors which relate to existing states, denoted in red. The number of factors re-used from previously calculated planning tree, denoted in orange. (b) The number of factors re-used from previously calculated planning tree, denoted in orange. The number of factors removed from the previously calculated planning tree, denoted in black.

Figure 7.3.5 presents the estimated route of the second experiment, coursing through 3 goals along 148 meters under both ML-BSP and iML-BSP. In this experiment, like in the former, both ML-BSP and iML-BSP chose the same optimal action sequence at each planning session. Similar to Figure 7.3.3, Figure 7.3.6 presents the timing results for the second experiment. Same as before we can

see that while the computation time related to the last horizon step is identical between ML-BSP and iML-BSP, there is a considerable time reduction in the computation time related to the first three horizon steps, from constituting 54% of the cumulative planning time in ML-BSP to just 25% in iML-BSP.

In a similar manner Figure 7.3.7, like Figure 7.3.4, presents the sum of factors related to the second experiment. As before, we can see that the orange line is quite close to the red line (i.e. upper bound for factor re-use), but there is a considerable number of factor to remove each planning step (black line in Figure 7.3.7b). As the second experiment provides us with the same insights over the comparison between ML-BSP and iML-BSP it essentially validates the results of the first experiment as well as the insights derived from it.

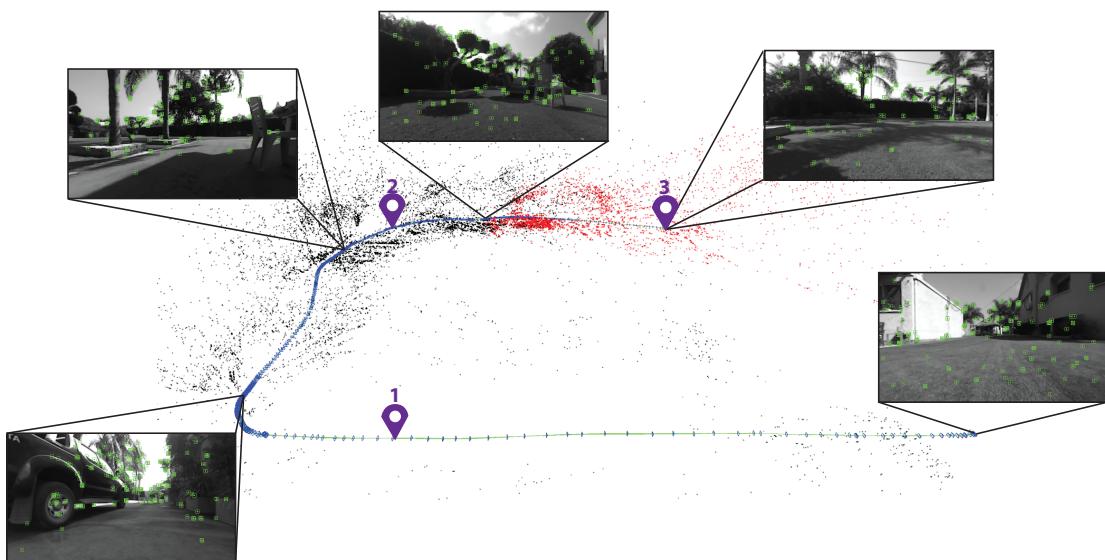


Figure 7.3.5: The second live experiment stretching across a 148 meter course, where the Pioneer robot was given 3 goals (numbered and denoted in purple) to reach. The state estimation of the robot in the form of keyframes (denoted in blue frames), robot trajectory (denoted in green) landmarks (denoted in black and red), and few snapshots along the rout containing the selected features.

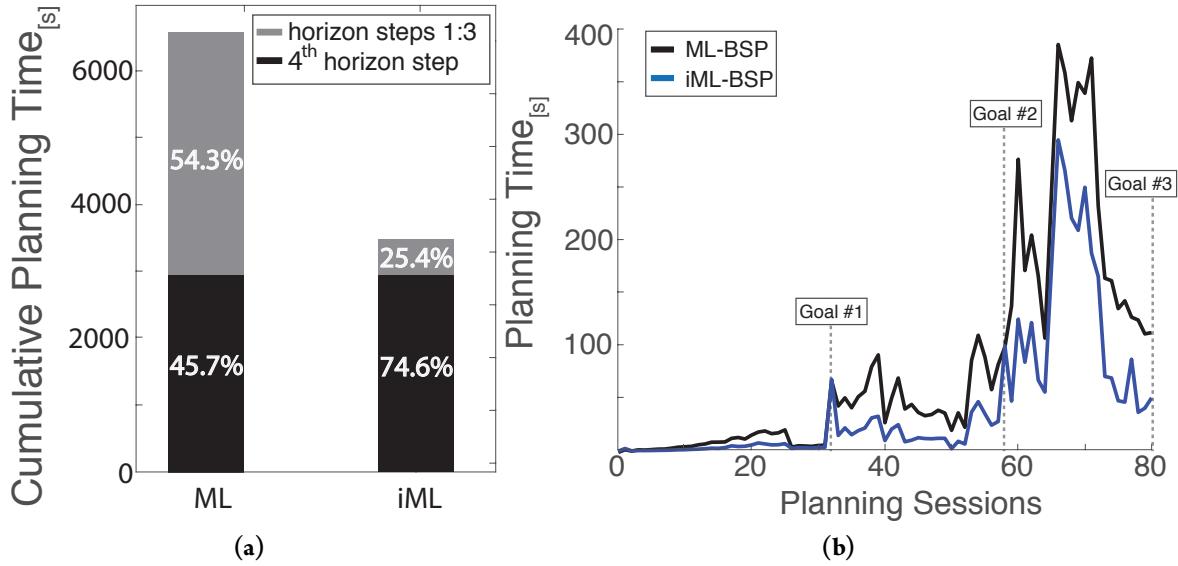


Figure 7.3.6: Planning time results of the second live experiment using the Pioneer robot (a) The cumulative planning time of both ML-BSP and iML-BSP, divided into relative contributions of the first three planning horizon steps (denoted in gray) and the last planning horizon step (denoted in black). (b) Per planning session comparison of the computation time. The goals are marked over the planning session performed after reaching them.

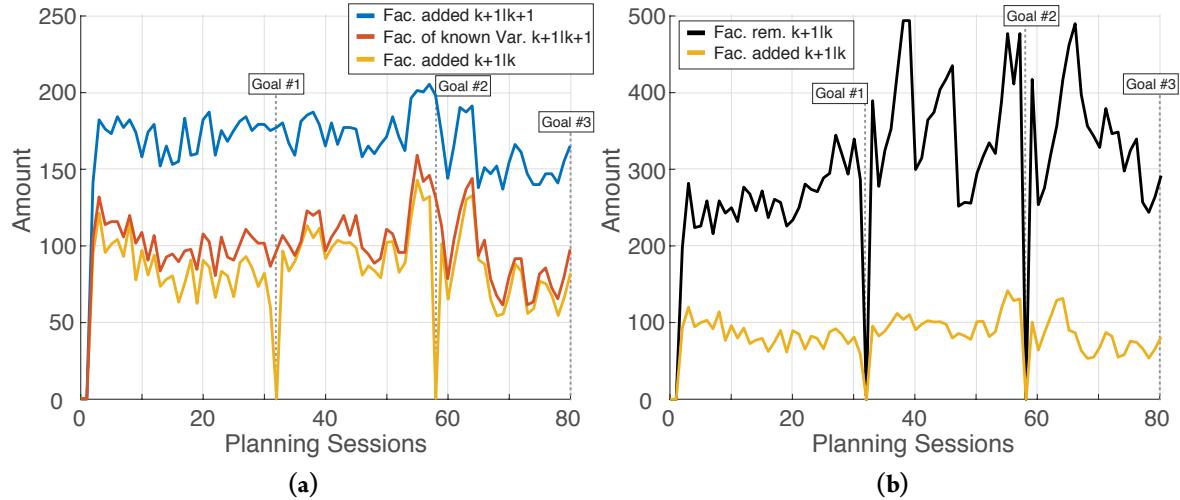


Figure 7.3.7: Number of factors involved in each planning session of the second live experiment. (a) The number of factors added in the last inference update at $k + 1|k + 1$, denoted in blue. The portion of aforementioned factors which relate to existing states, denoted in red. The number of factors re-used from previously calculated planning tree, denoted in orange. (b) The number of factors re-used from previously calculated planning tree, denoted in orange. The number of factors removed from the previously calculated planning tree, denoted in black.

The Cure for Boredom Is Curiosity.

There Is No Cure for Curiosity.

Ellen Parr

8

Closing Remarks

IF FORCED TO SUMMARIZE OUR ENTIRE RESEARCH WITH A SINGLE CRYPTIC SENTENCE, ONE COULD SAY THAT WE STARTED WITH TEARING DOWN A WALL AND ENDED UP WITH BUILDING TWO NEW BRIDGES.

Inference and decision making under uncertainty or inference and BSP are core processes in both AS and AI. In-spite of the obvious similarities between inference and BSP, several research efforts in the field, and the fact that in the human brain they are tightly entwined, they are still being treated as two separate processes. Our research tears down the wall between inference and BSP by introducing the joint inference and BSP framework - JIP. Under JIP both inference and planning are considered as part of the same plan-act-infer process, thus enabling to share similar calculations between them. JIP encapsulates the standard un-approximated Bayesian inference and BSP as well as enabling new symbiotic connections between the two.

The first symbiotic connection our work offered might seem as a paradigm shift from Bayesian inference, instead of updating the last available posterior belief with new information we update a future belief from a previous planning session. Given a future belief from precursory planning and newly acquired data, we appropriately update the former with the latter while taking into consideration data association inconsistencies which might occur. The resulting approach, RUB Inference,

saves valuable computation time in inference without affecting the estimation accuracy.

The second symbiotic connection our work offered allows to incrementally calculate the expectation of X-BSP through the selective re-use of previous planning sessions. The resulting approach, iX-BSP, saves valuable computation time in the creation of the planning tree without statistically affecting the solution accuracy. We also introduced a non-integral addition to iX-BSP, denoted as wildfire, allowing one to controllably trade accuracy for computational performance. As iX-BSP is equivalent to X-BSP, we believe that the existing approximations of X-BSP can benefit from the iX-BSP paradigm. To support our claim we introduced the common maximum likelihood approximation of X-BSP- ML-BSP, the iX-BSP paradigm. The resulting approach, iML-BSP, showed improved computation time when compared to the standard ML-BSP.

In addition to the formulation of our proposed methods, we compared each of them to the current top of the line, using both simulative and real-world data, thus demonstrating the claimed computational advantage as well as the accuracy performance.

Our research enables calculation re-use across inference and decision making under uncertainty, thus improving computation time without affecting accuracy. We strongly believe that both iX-BSP and RUB Inference can be further improved, by better utilizing the newly acquired connectivity enabled through JIP. We also believe that JIP has probably more to offer than just RUB Inference and iX-BSP, and we hope our work will pave the way to even more new symbiotic connections within JIP.

FUTURE RESEARCH DIRECTIONS

Although we can never predict all the possible future research that can be done based on our work, we can summarize the issues we encountered and would have chosen to address in case we had the time. All these future research directions will better improve RUB Inference and iX-BSP. Although none of them is directly suggesting a new connection in JIP, perhaps some of them will become the required inspiration for such connection.

- **Reducing factor eliminations by anticipating required ordering:** By predicting the required ordering and incrementally adjusting the joint state accordingly one can reduce the computational load of belief update both in inference and planning.
- **Integrating a more sophisticated mechanism for factor prediction:** In this work we used a straightforward mechanism for factor prediction that proved to have room for much improvement in two aspects.

- better prediction of existing states in sense of both observation and association.
- have the ability to predict new states.
- **Finding the Goldilocks conditions for ML-BSP with multiple actions and X-BSP**
- **Belief dataset management**, with the purpose of quickly locating the closest belief for re-use.
- **A procedure to analytically choose ε_c .** Or in other words, find mathematical connection between belief distance and the corresponding computational load of updating one of the to match the other.
- **Find wildfire bounds that are convenient enough for online usage.**
- **A procedure for dynamically choosing ε_{wf}** , in accordance with allowable accuracy sacrifice.
- **Introduce smart sampling**, instead of representing the distribution of the entire measurement space, focus on "interesting" segments of the measurement space.
- **Incrementally updating a previously calculated reward value** in an exact manner or controlled approximation.
- **Analytically compare the estimation variance of iX-BSP and X-BSP**
- **Relieve the assumption of α -Hölder reward** taken for wildfire bound calculation.
- **Extend iML-BSP to a rollout-based planner** with belief dependent reward.
- **Extend JIP to multi-robot framework**

Bibliography

- [1] A.-A. Agha-Mohammadi, S. Chakravorty, and N. M. Amato. FIRM: Sampling-based feedback motion planning under motion uncertainty and imperfect measurements. *Intl. J. of Robotics Research*, 33(2):268–304, 2014.
- [2] Ali-Akbar Agha-Mohammadi, Saurav Agarwal, Aditya Mahadevan, Suman Chakravorty, Daniel Tomkins, Jory Denny, and Nancy M Amato. Robust online belief space planning in changing environments: Application to physical mobile robots. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 149–156, 2014.
- [3] Ali-akbar Agha-mohammadi, Saurav Agarwal, Suman Chakravorty, and Nancy M Amato. Simultaneous localization and planning for physical mobile robots via enabling dynamic replanning in belief space. *arXiv preprint arXiv:1510.07380*, 2015.
- [4] Jonathan Binney, Andreas Krause, and Gaurav S Sukhatme. Optimizing waypoints for monitoring spatiotemporal phenomena. *Intl. J. of Robotics Research*, 32(8):873–888, 2013.
- [5] A. Bry and N. Roy. Rapidly-exploring random belief trees for motion planning under uncertainty. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 723–730, 2011.
- [6] L. Carlone, A. Censi, and F. Dellaert. Selecting good measurements via l_1 relaxation: A convex approach for robust estimation over graphs. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pages 2667–2674. IEEE, 2014.
- [7] Stephen M Chaves and Ryan M Eustice. Efficient planning with the Bayes tree for active SLAM. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 4664–4671. IEEE, 2016.
- [8] Gabriele Costante, Christian Forster, Jeffrey Delmerico, Paolo Valigi, and Davide Scaramuzza. Perception-aware path planning. *arXiv preprint arXiv:1605.04151*, 2016.
- [9] Rémi Coulom. Efficient selectivity and backup operators in monte-carlo tree search. In *International conference on computers and games*, pages 72–83. Springer, 2006.
- [10] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2006.
- [11] A. Cunningham, V. Indelman, and F. Dellaert. DDF-SAM 2.0: Consistent distributed smoothing and mapping. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, Karlsruhe, Germany, May 2013.

- [12] A.J. Davison, I. Reid, N. Molton, and O. Stasse. MonoSLAM: Real-time single camera SLAM. *IEEE Trans. Pattern Anal. Machine Intell.*, 29(6):1052–1067, Jun 2007.
- [13] F. Dellaert. Factor graphs and GTSAM: A hands-on introduction. Technical Report GT-RIM-CP&R-2012-002, Georgia Institute of Technology, September 2012.
- [14] F. Dellaert and M. Kaess. Square Root SAM: Simultaneous localization and mapping via square root information smoothing. *Intl. J. of Robotics Research*, 25(12):1181–1203, Dec 2006.
- [15] Dominik Maria Endres and Johannes E Schindelin. A new metric for probability distributions. *IEEE Transactions on Information theory*, 2003.
- [16] R.M. Eustice, H. Singh, and J.J. Leonard. Exactly sparse delayed-state filters for view-based SLAM. *IEEE Trans. Robotics*, 22(6):1100–1114, Dec 2006.
- [17] E. I. Farhi and V. Indelman. Towards efficient inference update through planning via jip - joint inference and belief space planning. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2017.
- [18] E. I. Farhi and V. Indelman. ix-bsp: Belief space planning through incremental expectation. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, May 2019.
- [19] E. I. Farhi and V. Indelman. Tear down that wall: Calculation reuse across inference and belief space planning. In *Toward Online Optimal Control of Dynamic Robots, Workshop in conjunction with IEEE International Conference on Robotics and Automation (ICRA)*, May 2019.
- [20] Neha P Garg, David Hsu, and Wee Sun Lee. Despot-a: Online pomdp planning with large state and observation spaces. In *Robotics: Science and Systems (RSS)*, 2019.
- [21] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun. Vision meets robotics: The kitti dataset. *International Journal of Robotics Research (IJRR)*, 2013.
- [22] Mohammad Ghavamzadeh, Shie Mannor, Joelle Pineau, Aviv Tamar, et al. Bayesian reinforcement learning: A survey. *Foundations and Trends® in Machine Learning*, 8(5-6):359–483, 2015.
- [23] David Gilbarg and Neil S Trudinger. *Elliptic partial differential equations of second order*. Springer, 2015.
- [24] Peter W Glynn and Donald L Iglehart. Importance sampling for stochastic simulations. *Management Science*, 35(11):1367–1392, 1989.
- [25] G.H. Golub and C.F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, third edition, 1996.
- [26] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.
- [27] Simon S Haykin. *Kalman filtering and neural networks*. Wiley Online Library, 2001.

- [28] G. A. Hollinger and G. S. Sukhatme. Sampling-based robotic information gathering algorithms. *Intl. J. of Robotics Research*, pages 1271–1287, 2014.
- [29] V. Indelman, L. Carlone, and F. Dellaert. Planning in the continuous domain: a generalized belief space approach for autonomous navigation in unknown environments. *Intl. J. of Robotics Research*, 34(7):849–882, 2015.
- [30] V. Indelman, E. Nelson, J. Dong, N. Michael, and F. Dellaert. Incremental distributed inference from arbitrary poses and unknown data association: Using collaborating robots to establish a common reference. *IEEE Control Systems Magazine (CSM), Special Issue on Distributed Control and Estimation for Robotic Vehicle Networks*, 36(2):41–74, 2016.
- [31] Harold Jeffreys. An invariant form for the prior probability in estimation problems. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 186(1007):453–461, 1946.
- [32] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1):99–134, 1998.
- [33] M. Kaess, A. Ranganathan, and F. Dellaert. iSAM: Incremental smoothing and mapping. *IEEE Trans. Robotics*, 24(6):1365–1378, Dec 2008.
- [34] M. Kaess, V. Ila, R. Roberts, and F. Dellaert. The Bayes tree: Enabling incremental reordering and fluid relinearization for online mapping. Technical Report MIT-CSAIL-TR-2010-021, Computer Science and Artificial Intelligence Laboratory, MIT, Jan 2010.
- [35] M. Kaess, V. Ila, R. Roberts, and F. Dellaert. The Bayes tree: An algorithmic foundation for probabilistic robot mapping. In *Intl. Workshop on the Algorithmic Foundations of Robotics*, Dec 2010.
- [36] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert. iSAM2: Incremental smoothing and mapping using the Bayes tree. *Intl. J. of Robotics Research*, 31:217–236, Feb 2012.
- [37] S. Karaman and E. Frazzoli. Sampling-based algorithms for optimal motion planning. *Intl. J. of Robotics Research*, 30(7):846–894, 2011.
- [38] Sertac Karaman, Matthew R Walter, Alejandro Perez, Emilio Frazzoli, and Seth Teller. Anytime motion planning using the rrt*. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*. IEEE, 2011.
- [39] L.E. Kavraki, P. Svestka, J.-C. Latombe, and M.H. Overmars. Probabilistic roadmaps for path planning in high-dimensional configuration spaces. *IEEE Trans. Robot. Automat.*, 12(4):566–580, 1996.
- [40] A. Kim and R. M. Eustice. Active visual SLAM for robotic area coverage: Theory and experiment. *Intl. J. of Robotics Research*, 34(4-5):457–475, 2014.

- [41] Marin Kobilarov, Duy-Nguyen Ta, and Frank Dellaert. Differential dynamic programming for optimal estimation. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 863–869. IEEE, 2015.
- [42] D. Kopitkov and V. Indelman. No belief propagation required: Belief space planning in high-dimensional state spaces via factor graphs, matrix determinant lemma and re-use of calculation. *Intl. J. of Robotics Research*, 36(10):1088–1130, August 2017.
- [43] D. Kopitkov and V. Indelman. General purpose incremental covariance update and efficient belief space planning via factor-graph propagation action tree. *Intl. J. of Robotics Research*, 2019.
- [44] F.R. Kschischang, B.J. Frey, and H-A. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Trans. Inform. Theory*, 47(2):498–519, February 2001.
- [45] H. Kurniawati, D. Hsu, and W. S. Lee. SARSOP: Efficient point-based POMDP planning by approximating optimally reachable belief spaces., 2008.
- [46] Hanna Kurniawati and Vinay Yadav. An online pomdp solver for uncertainty planning in dynamic environment. In *Robotics Research*, pages 611–629. Springer, 2016.
- [47] S. M. LaValle and J. J. Kuffner. Randomized kinodynamic planning. *Intl. J. of Robotics Research*, 20(5):378–400, 2001.
- [48] David G Lowe. Distinctive image features from scale-invariant keypoints. *Intl. J. of Computer Vision*, 60(2):91–110, 2004.
- [49] Yuanfu Luo, Haoyu Bai, David Hsu, and Wee Sun Lee. Importance sampling for online planning under uncertainty. *IJRR*, page 0278364918780322, 2016.
- [50] Yuanfu Luo, Haoyu Bai, David Hsu, and Wee Sun Lee. Importance sampling for online planning under uncertainty. *Intl. J. of Robotics Research*, 38(2-3):162–181, 2019.
- [51] Kai-Chieh Ma, Lantao Liu, and Gaurav S Sukhatme. Informative planning and online learning with sparse gaussian processes. *arXiv preprint arXiv:1609.07560*, 2016.
- [52] Arakaparampil M Mathai and Serge B Provost. *Quadratic forms in random variables: theory and applications*. Dekker, 1992.
- [53] Michael J. Milford and Gordon F. Wyeth. Mapping a suburb with a single camera using a biologically inspired SLAM system. *IEEE Trans. Robotics*, 24(5):1038–1053, October 2008.
- [54] E. Olson and P. Agarwal. Inference on networks of mixtures for robust robot mapping. *Intl. J. of Robotics Research*, 32(7):826–840, 2013.
- [55] C. Papadimitriou and J. Tsitsiklis. The complexity of markov decision processes. *Mathematics of operations research*, 12(3):441–450, 1987.

- [56] S. Pathak, A. Thomas, and V. Indelman. A unified framework for data association aware robust belief space planning and perception. *Intl. J. of Robotics Research*, 32(2-3):287–315, 2018.
- [57] J. Pineau, G. J. Gordon, and S. Thrun. Anytime point-based approximations for large POMDPs. *J. of Artificial Intelligence Research*, 27:335–380, 2006.
- [58] R. Platt, R. Tedrake, L.P. Kaelbling, and T. Lozano-Pérez. Belief space planning assuming maximum likelihood observations. In *Robotics: Science and Systems (RSS)*, pages 587–593, Zaragoza, Spain, 2010.
- [59] S. Prentice and N. Roy. The belief roadmap: Efficient planning in belief space by factoring the covariance. *Intl. J. of Robotics Research*, 28(11-12):1448–1465, 2009.
- [60] Elizabeth Race, Margaret M Keane, and Mieke Verfaellie. Medial temporal lobe damage causes deficits in episodic memory and episodic future thinking not attributable to deficits in narrative construction. *Journal of Neuroscience*, 31(28):10262–10269, 2011.
- [61] Charles Richter, William Vega-Brown, and Nicholas Roy. Bayesian learning for safe high-speed navigation in unknown environments. In *Proc. of the Intl. Symp. of Robotics Research (ISRR)*, 2015.
- [62] Daniel L Schacter and Donna Rose Addis. On the nature of medial temporal lobe contributions to the constructive simulation of future events. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 364(1521):1245–1253, 2009.
- [63] Daniel L Schacter, Donna Rose Addis, and Randy L Buckner. Remembering the past to imagine the future: the prospective brain. *Nature Reviews Neuroscience*, 8(9):657–661, 2007.
- [64] David Silver and Joel Veness. Monte-carlo planning in large pomdps. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2164–2172, 2010.
- [65] C. Stachniss, D. Haehnel, and W. Burgard. Exploration with active loop-closing for FastSLAM. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2004.
- [66] N. Sunderhauf and P. Protzel. Towards a robust back-end for pose graph slam. In *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pages 1254–1261. IEEE, 2012.
- [67] Duy-Nguyen Ta, Marin Kobilarov, and Frank Dellaert. A factor graph approach to estimation and model predictive control on unmanned aerial vehicles. In *International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 181–188. IEEE, 2014.
- [68] S. Thrun, Y. Liu, D. Koller, A.Y. Ng, Z. Ghahramani, and H. Durrant-Whyte. Simultaneous localization and mapping with sparse extended information filters. *Intl. J. of Robotics Research*, 23(7-8):693–716, 2004.
- [69] Emanuel Todorov. General duality between optimal control and estimation. In *IEEE Conference on Decision and Control*, pages 4286–4292. IEEE, 2008.

- [70] Marc Toussaint. Robot trajectory optimization using approximate inference. In *Intl. Conf. on Machine Learning (ICML)*, pages 1049–1056. ACM, 2009.
- [71] Marc Toussaint and Amos Storkey. Probabilistic inference for solving discrete and continuous state markov decision processes. In *Intl. Conf. on Machine Learning (ICML)*, pages 945–952. ACM, 2006.
- [72] R. Valencia, M. Morta, J. Andrade-Cetto, and J.M. Porta. Planning reliable paths with pose SLAM. *IEEE Trans. Robotics*, pages 1050–1059, 2013.
- [73] J. Van Den Berg, S. Patil, and R. Alterovitz. Motion planning under uncertainty using iterative local optimization in belief space. *Intl. J. of Robotics Research*, 31(11):1263–1278, 2012.
- [74] Eric Veach and Leonidas J Guibas. Optimally combining sampling techniques for monte carlo rendering. In *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 419–428. ACM, 1995.
- [75] Max A Woodbury. Inverting modified matrices. *Memorandum report*, 42(106):336, 1950.
- [76] Nan Ye, Adhiraj Somani, David Hsu, and Wee Sun Lee. Despot: Online pomdp planning with regularization. *JAIR*, 58:231–266, 2017.

*Progress is man's ability
to complicate simplicity.*

Thor Heyerdahl

A

Inference as a Graphical Model

The inference problem can be naturally represented and efficiently solved using graphical models such as factor graph (FG) [44] and Bayes tree (BT) [34]. Since FG and BT graphical models pose key components in the suggested paradigm, the theoretical foundation is supplied next. We use Figure 4.6.1 as illustration to belief representation in graphical models. Figures 4.6.1a and 4.6.1b are FG representations for the beliefs $b(X_{k+1|k})$ and $b(X_{k+1|k+1})$, respectively. BT representation of the belief is obtained through an elimination process, Figure 4.6.1d presents the BT of $b[X_{k+1|k}]$ for the elimination order $x_o \dots l_i \rightarrow x_{k-1} \rightarrow x_k \rightarrow l_j \rightarrow x_{k+1}$, while Figure 4.6.1e presents the BT of $b[X_{k+1|k+1}]$ for the elimination order $x_o \dots l_i \rightarrow x_{k-1} \rightarrow x_k \rightarrow l_j \rightarrow l_r \rightarrow x_{k+1}$.

A FG is a bipartite graph with two node types, factor nodes $\{f_i\}$ and variable nodes $\{\theta_j\} \in \Theta$. All nodes are connected through edges $\{e_{ij}\}$, which are always between factor nodes to variable nodes. A factor graph defines the factorization of a certain function $g(\Theta)$ as

$$g(\Theta) = \prod_i f_i(\Theta_i), \quad (\text{A.1})$$

where Θ_i is the set of variables $\{\theta_j\}$ connected to the factor f_i through the set of edges $\{e_{ij}\}$. After substituting Θ with our joint state X and the factors $\{f_i\}$ with the conditional probabilities from Eq. (??) we receive the definition of the belief $b(X_{t|k})$ in a FG representation.

Through bipartite elimination game, a FG can be converted into a BN, this elimination is required for solving the Inference problem (as shown in [36]). After eliminating all variables the BN pdf can be defined by a product of conditional probabilities,

$$P(\Theta) = \prod_j P(\Theta_j | S_j), \quad (\text{A.2})$$

where S_j is addressed as the *separator* of Θ_j , i.e. the set of variables that are directly connected to Θ_j .

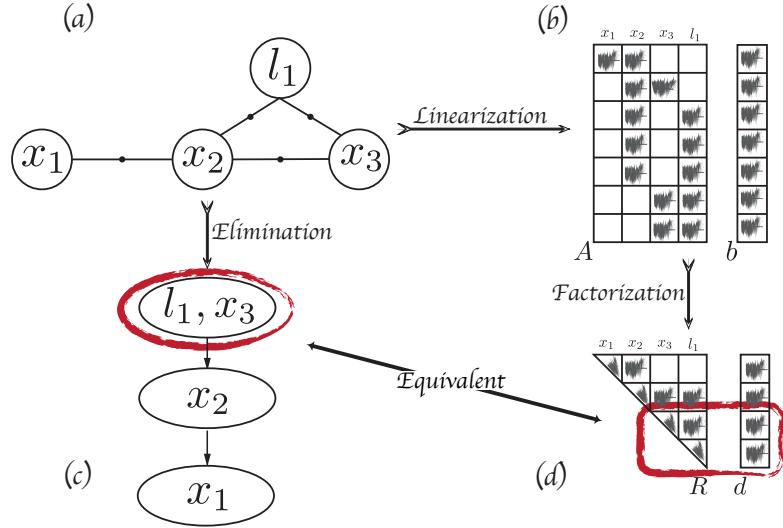


Figure A.1: The relations between different problem representations. (a) Factor graph (b) Jacobian matrix A with RHS vector b (c) Bayes Tree (d) Factorized Jacobian matrix R with equivalent RHS vector d .

In order to ease optimization and marginalization, a BT can be used [36]. By converting the BN to a directed tree, where the nodes represent *cliques* $\{C_r\}$, we receive a directed graphical model that encodes a factored pdf. Bayes Tree is defined using a conditional density per each node.

$$P(\Theta) = \prod_r P(F_r | S_r), \quad (\text{A.3})$$

where S_r is the separator, defined by the intersection $C_r \cap \Pi_r$ of the clique C_r and the parent clique Π_r . The complement to the variables in the clique C_r is denoted as F_r , the *frontal variables*. Each clique is therefore written in the form $C_r = F_r : S_r$.

The correspondence between matrix and graphical representation is conveniently demonstrated in Figure A.1. The first rows of R are equivalent to the deepest cliques in the BT, when the last rows of R are equivalent to the root of the tree. The elimination order that created the BT is identical to the ordering of R state vector, and fill-ins in R are equivalent to the connectivity of the corresponding BT.

If I were two-faced,
would I be wearing this one?

Abraham Lincoln

B

Derivation of Eq. (4.14)

In this appendix we complete the derivation of Eq. (4.14) from Eq. (4.13). Let us consider the NLS presented in Eq. (4.13)

$$X_{k|k}^* = \arg \min_{X_k} \|x_o - x_o^*\|_{\Sigma_o}^2 + \sum_{i=1}^k \left[\|x_i - f(x_{i-1}, u_{i-1|k})\|_{\Sigma_w}^2 + \sum_{j \in \mathcal{M}_i|k} \|z_{i|k}^j - h(x_i, l_j)\|_{\Sigma_v}^2 \right].$$

In general, the motion model $f(\cdot)$ and the measurement model $h(\cdot)$ are non-linear functions. A standard way to solve this problem is the Gauss-Newton method, where a single iteration involves linearizing about the last known estimate, calculating the delta around this linearization point, and updating the latter with the former. This process should be repeated until convergence.

We start by linearizing the terms in (4.13) using first order Taylor approximation around the best estimate we have for the joined state $\bar{X}_{k|k-1}$ which is the state estimate for time k before including measurements, i.e. $X_{k|k-1}^*$.

The prior term yields,

$$x_o - x_o^* = \bar{x}_o + \Delta x_o - x_o^* = \Delta x_o. \quad (\text{B.1})$$

The motion model term yields,

$$x_i - f(x_{i-1}, u_{i-1|k}) = \bar{x}_i - f(\bar{x}_{i-1}, u_{i-1|k}) - \Sigma_w^{-\frac{1}{2}} \mathcal{F}_i \begin{bmatrix} \Delta x_{i-1} \\ \Delta x_i \end{bmatrix} \quad (\text{B.2})$$

where $\Sigma_w^{-\frac{1}{2}} \mathcal{F}_i$ represents the Jacobian matrix of the motion model at time i , around the linearization

point $\bar{x}_{i-1:i}$. The measurement model term yields,

$$z_{i|k}^j - h(x_i, l_j) = z_{i|k}^j - h(\bar{x}_i, \bar{l}_j) - \Sigma_v^{-\frac{1}{2}} \mathcal{H}_{i,j} \begin{bmatrix} \Delta x_i \\ \Delta l_j \end{bmatrix} \quad (\text{B.3})$$

where $\Sigma_v^{-\frac{1}{2}} \mathcal{H}_{i,j}$ represents the jacobian matrix of the measurement model at time i around the linearization point $[\bar{x}_i, \bar{l}_j]^T$.

In order to re-write (4.13) into the common form of Least Squares $Ax = b$, we introduce Eqs. (B.1-B.3) back to (4.13),

$$\Delta X_{k|k}^* = \arg \min_{\Delta X_k} \left\| \Sigma_o^{-\frac{1}{2}} \Delta x_o \right\|^2 + \sum_{i=1}^k \left[\left\| \Sigma_w^{-\frac{1}{2}} \Delta x_i - \mathcal{F}_i \Delta x_{i-1} - \check{b}_i^{\mathcal{F}} \right\|^2 + \sum_{j \in \mathcal{M}_i|k} \left\| \mathcal{H}_{i,j} \begin{bmatrix} \Delta x_i \\ \Delta l_j \end{bmatrix} - \check{b}_i^{\mathcal{H}} \right\|^2 \right],$$

where the RHS terms $\check{b}_i^{\mathcal{F}}$ and $\check{b}_i^{\mathcal{H}}$ are given by

$$\check{b}_i^{\mathcal{F}} = \Sigma_w^{-\frac{1}{2}} (f(\bar{x}_{i-1}, u_{i-1|k}) - \bar{x}_i) \quad , \quad \check{b}_i^{\mathcal{H}} = \Sigma_v^{-\frac{1}{2}} (z_{i|k}^j - h(\bar{x}_i, \bar{l}_j)).$$

We now make use of the fact that the minimum sum of quadratic expressions is the minimum of each quadratic expression individually and is equal to zero. Thus enabling us to stack up all equations to form,

$$\Delta X_{k|k}^* = \arg \min_{\Delta X_k} \|A_{k|k} \Delta X_k - b_{k|k}\|^2,$$

where the Jacobian matrix and the RHS are given by,

$$A_{k|k} = \begin{bmatrix} \Sigma_o^{-\frac{1}{2}} \\ \mathcal{F}_{1:k|k} \\ \mathcal{H}_{1:k|k} \end{bmatrix} \quad , \quad b_{k|k} = \begin{bmatrix} o \\ \check{b}_{1:k|k}^{\mathcal{F}} \\ \check{b}_{1:k|k}^{\mathcal{H}} \end{bmatrix}.$$

*All generalizations are false,
including this one.*

Mark Twain

C

Non-zeros in Q Matrix

In this appendix we discuss the number of non zeros in the rotation matrix $Q_{k+1|k+1}$, in order to do so we first cover the creation of $Q_{k+1|k+1}$, and later get to an expression for the number of non zeros and analyze it to gain better understanding over the governing parameters. The rotation matrix $Q_{k+1|k+1}$ is created as part of the factorization of the Jacobian, designed to rotate the Jacobian into a square upper triangular form (e.g. Eqs. (4.17) and (4.25)). As such, we can deduce an expression for the number of non zeros in $Q_{k+1|k+1}$ as a function of the state size and the size of added factors, but first let us review how $Q_{k+1|k+1}$ is being created. Figure C.1a illustrates a simple example for the Jacobian matrix $A_{k+1|k+1}^R$, where the precursory factorized Jacobian is denoted by $R_{k|k}$, the newly added factors by $\mathcal{A}_{k+1|k+1}$ and the columns denote the different states. The involved variables in $\mathcal{A}_{k+1|k+1}$ are marked with light blue and orange. As can be deduced from Figure C.1a, the number $A_{k+1|k+1}^R$ columns equals the joint state size at time $k + 1$, and the number of $A_{k+1|k+1}^R$ rows equals the sum of the joint state size plus the number of $\mathcal{A}_{k+1|k+1}$ rows. The purpose of factorization is to rotate $A_{k+1|k+1}^R$ to a square upper triangular form without loosing information, i.e. so that $A_{k+1|k+1}^R {}^T A_{k+1|k+1}^R = R_{k+1|k+1} {}^T R_{k+1|k+1}$. While there are many different factorization algorithms, we would consider for simplicity without affecting generality the Given's Rotation (see [25]). Given's rotation creates $Q_{k+1|k+1}$ by a series of simple one cell rotations. For the simple case presented in Figure C.1, two rotations are required as presented in Figure C.1b. First the left-most non zero entry in $\mathcal{A}_{k+1|k+1}$, denoted by light blue, is addressed. The appropriate rotation matrix, consists of two off-diagonal non zeros denoted by dark blue, is denoted in Figure C.1b as the light blue $Q_{k+1|k+1}$. Next we are left to address the orange non zero entry in $\mathcal{A}_{k+1|k+1}$, while its appropriate rotation matrix, also consists of two off-diagonal non zeros denoted by dark red, is denoted in Figure C.1b as the orange $Q_{k+1|k+1}$. From Figure C.1b we can see that each sequential rotation matrix has the same number of non zeros, $\text{diag}(Q_{k+1|k+1}) + 2$, but due to the multiplication between them we get more non zeros as seen in Figure C.1c. For some intuition we marked the entries of the equivalent $Q_{k+1|k+1}$ presented in Figure C.1c, in accordance to

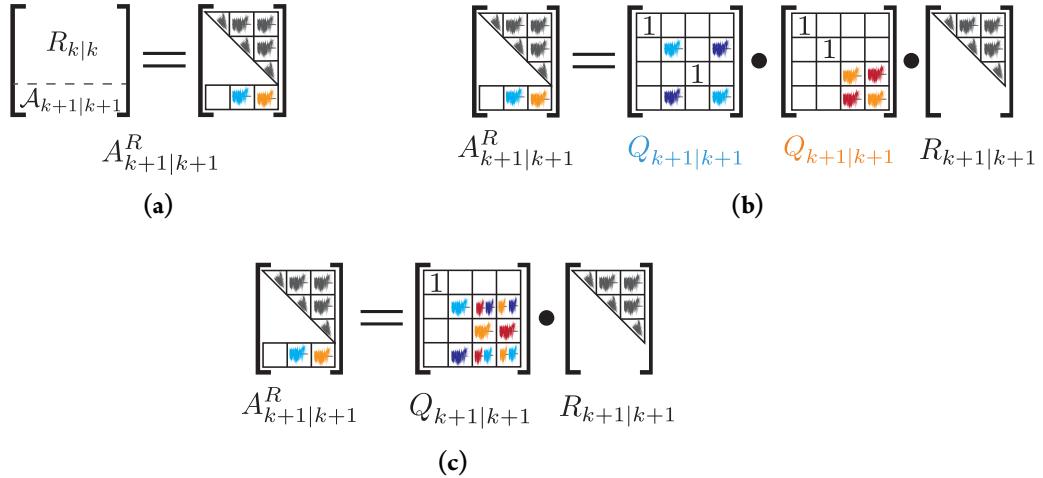


Figure C.1: (a) A Jacobian matrix at time $k + 1$, consisting of the previously factorized Jacobian from time k and the linearized newly added factor from time $k + 1$. The RHS visualize the non zeros of the aforementioned Jacobian. (b) Visualizing the factorization procedure of the Jacobian in (a) using two Given's rotation matrices. The light and dark colors represent the cosine and sine values respectively, attributed to each of the original non zeros in (a). (c) Visualizing the non zeros in the rotation matrix required to factorize the Jacobian (a), this rotation matrix is the product of the two matrices in (b), as such the non zeros are affected by the cosine and sine values in (b).

the color coding in Figure C.1b.

Now that we understand that the number of non zeros in $Q_{k+1|k+1}$ is affected by the size of the joint state, the size of the newly added factors and the location of the left-most involved state, we are in position to formulate the expression for the number of non zeros in $Q_{k+1|k+1}$. We invite the reader to refresh his memory regarding the notations used in this analysis using Figure 4.4.1, nevertheless all notations are also defined here.

Let j be the column index of the left-most involved state in the newly added factors $\mathcal{A}_{k+1|k+1}$, n^s be the size of the state vector (i.e. number of states multiplied by the state dimension), and n^f be the number of rows of $\mathcal{A}_{k+1|k+1}$ (i.e. number of factors multiplied by the factors' dimension). The number of non zeros can be defined as the sum of three values: the number of diagonal entries equal to 1, the contribution of the Jacobian line with the left-most state to the non zeros, and the contribution of the rest of the Jacobian lines. We will now calculate each of them.

As can be seen from Figure C.1b, the incremental rotation matrix (i.e. colored $Q_{k+1|k+1}$) created to rotate an entry in the i^{th} column, would have $i - 1$ diagonal entries equal to 1. Since the left-most state is located in the j^{th} column the number of diagonal entries equal to 1 in $Q_{k+1|k+1}$ would be

$$j-1, \quad \quad \quad (\text{C.1})$$

where j is bounded by the size of the state such that

$$j \in [1, n^s]. \quad (\text{C.2})$$

The rotation matrix Q for rotating an entire Jacobian row located in the i^{th} , with a left-most non zero located in the j^{th} column, would have non zeros in the i^{th} row from column j up to the last column and an fully dense upper triangle of non zeros over the same columns. This means that rotating the row

with the left-most index in column j would contribute the following number of non zeros to $Q_{k+1|k+1}$

$$\frac{n^2 - n}{2} + n + n - 1 = \frac{n^2}{2} + \frac{3}{2}n - 1, \quad (\text{C.3})$$

where n is defined by

$$n = n^s + n^f - j + 1. \quad (\text{C.4})$$

Assuming the left-most state in the Jacobian is located in the j^{th} column, rotating the rest of the rows of the Jacobian will only add non zeros at the appropriate rows in Q , without adding new non zeros to the appropriate upper triangle. The remaining $n^f - 1$ rows will contribute to $Q_{k+1|k+1}$ the following number of non zeros

$$\sum_{i=1}^{n^f-1} (n^s + n^f - i + 1) = (n^f - 1) (n^s + n^f + 1) - \frac{n^f(n^f - 1)}{2} = (n^f - 1)(n^s + \frac{n^f}{2} + 1). \quad (\text{C.5})$$

Evidently, the number of non zeros in $Q_{k+1|k+1}$ is given by

$$\underbrace{j - 1}_{i} + \underbrace{\frac{n^2}{2} + \frac{3}{2}n - 1}_{ii} + \underbrace{(n^f - 1)(n^s + \frac{n^f}{2} + 1)}_{iii}, \quad (\text{C.6})$$

where term (i) in Eq. (C.6) denotes the number of diagonal entries equal 1, term (ii) in Eq. (C.6) denotes the non zeros added after factorizing the factor with the left-most state j , term (iii) in Eq. (C.6) denotes the non zeros added after factorizing the rest of the factors. It is worth stressing that the value of j is acutely affected by the ordering of the joint state vector. For better ordering, j would receive larger values.

Now that we have an expression to the number of non zeros in $Q_{k+1|k+1}$, we would like to investigate which part of it is dominant. In the sequel we reformulate Eq. (C.6) into a sum of quadratic terms, and then find conditions to determine which term is dominant.

We start by introducing (C.4) into Eq. (C.6) and using simple arithmetics in order to get quadratic forms,

$$\frac{1}{2}n^{s2} + n^{f2} + 2n^sn^f + \frac{3}{2}n^s - n^sj + 3n^f - n^fj + \frac{1}{2}j^2 - \frac{3}{2}j - 1 \quad (\text{C.7})$$

$$\frac{1}{2} \left(n^s + n^f - j + \frac{3}{2} \right)^2 + \frac{1}{2}n^{f2} + \frac{3}{2}n^f + n^sn^f - \frac{17}{8} \quad (\text{C.8})$$

$$\underbrace{\frac{1}{2} \left(n^s + n^f - j + \frac{3}{2} \right)^2}_a + \underbrace{\frac{1}{2} \left(n^f + \frac{3}{2} \right)^2}_b + \underbrace{n^sn^f - \frac{26}{8}}_c. \quad (\text{C.9})$$

We have three candidates to be the dominant part of Eq. (C.9), denoted by terms (a) (b) and (c). Let us examine them to decide which is the dominant one and under what conditions. First we can see that term (b) in (C.9) is a special case of term (a) in (C.9) where $j = n^s$. Subsequently we are left

with comparing terms (a) and (c) in (C.9), i.e. we would like to check when

$$\left(n^s + n^f - j + \frac{3}{2}\right)^2 > n^s n^f, \quad (\text{C.10})$$

we define $a \triangleq n^s - j + \frac{3}{2}$ and get

$$a^2 + 2an^f + n^{f^2} - n^s n^f > 0. \quad (\text{C.11})$$

So we can say term (a) in (C.9) is bigger than term (c) in (C.9) when

$$\left(n^s - j + \frac{3}{2} > \sqrt{n^s n^f} - n^f\right) \cup \left(n^s - j + \frac{3}{2} < -\sqrt{n^s n^f} - n^f\right). \quad (\text{C.12})$$

Considering Eq. (C.2), we can dismiss $n^s - j + \frac{3}{2} < -\sqrt{n^s n^f} - n^f$ because the smallest the LHS can be is $\frac{3}{2}$, which will always be greater than the non positive number $-\sqrt{n^s n^f} - n^f$, so the condition on j so that term (a) is the dominant part of (C.9) is

$$n^s - \sqrt{n^s n^f} + n^f + \frac{3}{2} > j, \quad (\text{C.13})$$

which after considering Eq. (C.2) is true if and only if

$$-\sqrt{n^s n^f} + n^f + \frac{3}{2} > 0. \quad (\text{C.14})$$

We can now solve the aforementioned to get a condition to assure (C.13) holds,

$$n^f + \frac{3}{2} > \sqrt{n^s n^f} \quad (\text{C.15})$$

$$n^{f^2} - (n^s - 3)n^f + \frac{9}{4} > 0 \quad (\text{C.16})$$

$$\left(n^f > \frac{n^s - 3}{2} + \frac{\sqrt{n^{s^2} - 6n^s}}{2}\right) \cup \left(0 < n^f < \frac{n^s - 3}{2} - \frac{\sqrt{n^{s^2} - 6n^s}}{2}\right), \quad (\text{C.17})$$

where $\frac{n^s - 3}{2} - \frac{\sqrt{n^{s^2} - 6n^s}}{2}$ is non negative $\forall n^s$, and both conditions are defined for $n^s \geq 6$ which for a 6DOF problem means a single state. For a value of $n^s = 6$, $\frac{n^s - 3}{2} - \frac{\sqrt{n^{s^2} - 6n^s}}{2} = 1.5$ and for $n^s = 7$, $\frac{n^s - 3}{2} - \frac{\sqrt{n^{s^2} - 6n^s}}{2} < 1$ so effectively this condition is irrelevant $\forall n^s \neq 6$, so we are left with

$$\left(n^f > \frac{n^s - 3}{2} + \frac{\sqrt{n^{s^2} - 6n^s}}{2}\right) \cup (n^s \geq 6). \quad (\text{C.18})$$

Although this is the exact condition to insure term (a) is the dominant part of (C.9), in order to provide a more convenient condition we suggest an upper bound in the simple form of $n^f > n^s$.

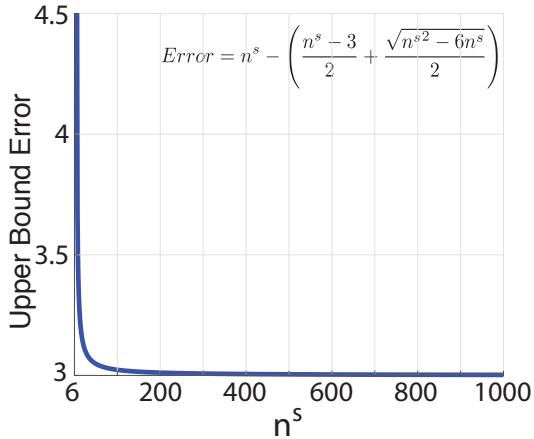


Figure C.2: Illustrating the effectiveness of the bound for $\frac{n^s - 3}{2} + \frac{\sqrt{n^{s2} - 6n^s}}{2}$ in the form of the error between the two as a function of different state sizes n^s .

Figure C.2 illustrates the effectiveness of the suggested bound in the form of the distance

$$\text{Error} = n^s - \frac{n^s - 3}{2} + \frac{\sqrt{n^{s2} - 6n^s}}{2}. \quad (\text{C.19})$$

For $n^s = 6$ the distance is 4.5, and for $n^s = 8$ it is already 3.5, which makes this bound very attractive for simplicity reasons.

To conclude, term (a) is the dominant part of (C.9) if and only if the following holds

$$\left(n^f > \frac{n^s - 3}{2} + \frac{\sqrt{n^{s2} - 6n^s}}{2} \right) \cap (n^s \geq 6), \quad (\text{C.20})$$

or for simpler upper bound

$$n^f > n^s \geq 6. \quad (\text{C.21})$$

Otherwise, term (c) is the dominant part of (C.9), i.e. given simply the size of the state and the number of rows of the newly added factors we can determine what will be the governing expression for determining the number of non zeros in $Q_{k+1|k+1}$.

*Between two evils,
I always pick the one I never tried before.*

Mae West

D

Multiple Importance Sampling

Let us assume we wish to express expectation over some function $f(x)$ with respect to distribution $p(x)$, by sampling x from a different distribution $q(x)$,

$$\mathbb{E}_p f(x) = \int f(x) \cdot p(x) dx = \int \frac{f(x) \cdot p(x)}{q(x)} q(x) dx = \mathbb{E}_q \left(\frac{f(x) \cdot p(x)}{q(x)} \right). \quad (\text{D.1})$$

Eq. (D.1) presents the basic importance sampling problem, where \mathbb{E}_q denotes expectation for $x \sim q(x)$. The probability ratio between the nominal distribution p and the importance sampling distribution q is usually referred to as the likelihood ratio. Our problem is more complex, since our samples are potentially taken from M different distributions while $M \in [1, (n_x \cdot n_z)^L]$, i.e. a multiple importance sampling problem

$$\mathbb{E}_p f(x) = \tilde{\mu}(x) \sim \sum_{m=1}^M \frac{1}{n_m} \sum_{i=1}^{n_m} w_m(x_{im}) \frac{f(x_{im}) p(x_{im})}{q_m(x_{im})}, \quad (\text{D.2})$$

where $w_m(\cdot)$ are weight functions satisfying $\sum_{m=1}^M w_m(x) = 1$, n_m denotes the number of samples from the M_{th} distribution. For $q_m(x) > 0$ whenever $w_m(x)p(x)f(x) \neq 0$, Eq. (D.2) forms an unbiased estimator

$$\mathbb{E} [\tilde{\mu}(x)] = \sum_{m=1}^M \mathbb{E}_{q_m} \left[\frac{1}{n_m} \sum_{i=1}^{n_m} w_m(x_{im}) \frac{f(x_{im}) p(x_{im})}{q_m(x_{im})} \right] = \tilde{\mu}(x). \quad (\text{D.3})$$

Although there are numerous options for weight functions satisfying $\sum_{m=1}^M w_m(x) = 1$, we chose to consider the Balance Heuristic [74], considered to be nearly optimal in the sense of estimation vari-

ance [74, Theorem 1],

$$w_m(x) = w_m^{BH}(x) = \frac{n_m q_m(x)}{\sum_{s=1}^M n_s q_s(x)}. \quad (\text{D.4})$$

Using (D.4) in (D.2) produces the multiple importance sampling with the balance heuristic

$$\mathbb{E}_p f(x) \sim \frac{1}{n} \sum_{m=1}^M \sum_{i=1}^{n_m} \frac{p(x_{im})}{\sum_{s=1}^M \frac{n_s}{n} q_s(x_{im})} f(x_{im}). \quad (\text{D.5})$$

*I refuse to join any club
that would have me as a member.*

Groucho Marx

E

The $\mathbb{D}_{\sqrt{J}}$ Distance

In this work we are required to make use of a probability density function (pdf) distance. After some consideration we chose to use $\mathbb{D}_{\sqrt{J}}$, which is a variant of the Jeffreys divergence \mathbb{D}_J first suggested in [31],

$$\mathbb{D}_{\sqrt{J}}(P, Q) = \sqrt{\frac{1}{2}\mathbb{D}_J} = \sqrt{\frac{1}{2}\mathbb{D}_{KL}(P||Q) + \frac{1}{2}\mathbb{D}_{KL}(Q||P)}, \quad (\text{E.1})$$

where P and Q are probability density functions and $\mathbb{D}_{KL}(P||Q)$ is the Kullback-Leibler(KL) divergence.

The Kullback-Leibler(KL) divergence, sometime referred to as relative entropy, measures how well some distribution Q approximates distribution P , or in other words how much information will be lost if one considers distribution Q instead of P . The KL divergence is not a metric (asymmetric) and is given by

$$\mathbb{D}_{KL}(P||Q) = \int P \cdot \log \frac{P}{Q} = \mathbb{E}_P [\log P - \log Q]. \quad (\text{E.2})$$

From a view point of Bayesian Inference, as explained in [15], the $\mathbb{D}_{KL}(P||Q)$ metric can be interpreted as twice the expected information gain when deciding between P and Q given a uniform prior over them.

For the special case of Gaussian distributions, we can express $\mathbb{D}_{KL}(P||Q)$ and consequently also $\mathbb{D}_{\sqrt{J}}(P, Q)$ in terms of means and covariances. Let us consider two multivariate Gaussian distribu-

tions $P \sim \mathcal{N}(\mu_p, \Sigma_p)$ and $Q \sim \mathcal{N}(\mu_q, \Sigma_q)$ in \mathbb{R}^d .

$$\begin{aligned}
\mathbb{D}_{KL}(P||Q) &= \mathbb{E}_P [\log P - \log Q] \\
&= \frac{1}{2} \mathbb{E}_P \left[-\log |\Sigma_p| - (\mathbf{x} - \mu_p)^T \Sigma_p^{-1} (\mathbf{x} - \mu_p) + \log |\Sigma_q| + (\mathbf{x} - \mu_q)^T \Sigma_q^{-1} (\mathbf{x} - \mu_q) \right] \\
&= \frac{1}{2} \log \frac{|\Sigma_q|}{|\Sigma_p|} + \frac{1}{2} \mathbb{E}_P \left[-(\mathbf{x} - \mu_p)^T \Sigma_p^{-1} (\mathbf{x} - \mu_p) + (\mathbf{x} - \mu_q)^T \Sigma_q^{-1} (\mathbf{x} - \mu_q) \right] \\
&= \frac{1}{2} \log \frac{|\Sigma_q|}{|\Sigma_p|} + \frac{1}{2} \mathbb{E}_P \left[-\text{tr}(\Sigma_p^{-1} (\mathbf{x} - \mu_p)^T (\mathbf{x} - \mu_p)) + \text{tr}(\Sigma_q^{-1} (\mathbf{x} - \mu_q)^T (\mathbf{x} - \mu_q)) \right] \\
&= \frac{1}{2} \log \frac{|\Sigma_q|}{|\Sigma_p|} - \frac{1}{2} \text{tr}(\Sigma_p^{-1} \Sigma_p) + \frac{1}{2} \mathbb{E}_P \left[\text{tr} \left(\Sigma_q^{-1} (\mathbf{x} \mathbf{x}^T - 2\mathbf{x}\mu_q^T + \mu_q \mu_q^T) \right) \right] \\
&= \frac{1}{2} \log \frac{|\Sigma_q|}{|\Sigma_p|} - \frac{1}{2} d_p + \frac{1}{2} \text{tr} \left(\Sigma_q^{-1} (\Sigma_p + \mu_p \mu_p^T - 2\mu_p \mu_q^T + \mu_q \mu_q^T) \right) \\
&= \frac{1}{2} \left[\log \frac{|\Sigma_q|}{|\Sigma_p|} - d_p + \text{tr} \left(\Sigma_q^{-1} \Sigma_p \right) + \text{tr} \left(\Sigma_q^{-1} (\mu_p - \mu_q)^T (\mu_p - \mu_q) \right) \right] \\
&= \frac{1}{2} \left[\log \frac{|\Sigma_q|}{|\Sigma_p|} - d_p + \text{tr} \left(\Sigma_q^{-1} \Sigma_p \right) + (\mu_p - \mu_q)^T \Sigma_q^{-1} (\mu_p - \mu_q) \right]. \tag{E.3}
\end{aligned}$$

Substituting Eq. (E.3) in Eq. (E.1) we get the $\mathbb{D}_{\sqrt{J}}$ representation for the multivariate Gaussian case,

$$\begin{aligned}
\mathbb{D}_{\sqrt{J}}(P, Q) &= \\
&\sqrt{\frac{1}{4} \left[\log \frac{|\Sigma_q|}{|\Sigma_p|} - d_p - d_q + \text{tr} \left(\Sigma_q^{-1} \Sigma_p \right) + (\mu_p - \mu_q)^T \left[\Sigma_q^{-1} + \Sigma_p^{-1} \right] (\mu_p - \mu_q) + \log \frac{|\Sigma_p|}{|\Sigma_q|} + \text{tr} \left(\Sigma_p^{-1} \Sigma_q \right) \right]} \\
&= \frac{1}{2} \sqrt{(\mu_p - \mu_q)^T \left[\Sigma_q^{-1} + \Sigma_p^{-1} \right] (\mu_p - \mu_q) + \text{tr} \left(\Sigma_q^{-1} \Sigma_p \right) + \text{tr} \left(\Sigma_p^{-1} \Sigma_q \right) - d_p - d_q}. \tag{E.4}
\end{aligned}$$

*if you want a guarantee
buy a toaster.*

Clint Eastwood

F

Proof of Theorem 1.

Lemma 4. For any two distributions P and Q , and $\alpha \in (0, 1]$ the L_1^α distance is bounded by the KL divergence in the following manner

$$\|P - Q\|_1^\alpha \leq [2 \cdot \ln 2 \cdot \mathbb{D}_{KL}(P\|Q)]^{\frac{\alpha}{2}} \quad (\text{F.1})$$

Proof. Following Lemma 11.6.1 in [10],

$$\frac{1}{2 \cdot \ln 2} \|P - Q\|_1^2 \leq \mathbb{D}_{KL}(P\|Q), \quad (\text{F.2})$$

multiplying both sides by the positive constant $2 \cdot \ln 2$ and raising to the $\frac{\alpha}{2}$ power we get

$$\|P - Q\|_1^\alpha \leq [2 \cdot \ln 2 \cdot \mathbb{D}_{KL}(P\|Q)]^{\frac{\alpha}{2}}. \quad (\text{F.3})$$

□

Theorem 1 (Bounded reward difference). Let $r(b, u)$ be α -Hölder continuous with λ_α and $\alpha \in (0, 1]$. Let b and b' denote two beliefs. Then the difference between $r(b, u)$ and $r(b', u)$ is bounded by

$$|r(b, u) - r(b', u)| \leq (4 \cdot \ln 2)^{\frac{\alpha}{2}} \cdot \lambda_\alpha \cdot \mathbb{D}_{\sqrt{J}}(b, b'). \quad (\text{F.4})$$

where

$$\mathbb{D}_{\sqrt{J}}(b, b') = \sqrt{\frac{1}{2} \mathbb{D}_{KL}(b\|b') + \frac{1}{2} \mathbb{D}_{KL}(b'\|b)}, \quad (\text{F.5})$$

and $\mathbb{D}_{KL}(\cdot)$ is the KL divergence.

Proof. The reward function $r(b, u)$ is α -Hölder continuous with λ_α and α so following Eq. (4.4) in

[23],

$$|r(b, u) - r(b', u)| \leq \lambda_a \cdot \|b - b'\|_1^a, \quad (\text{F.6})$$

using Lemma 4 we can rewrite the bound as

$$|r(b, u) - r(b', u)| \leq (4 \cdot \ln 2)^{\frac{a}{2}} \cdot \lambda_a \cdot \left(\frac{1}{2} \mathbb{D}_{KL}(b\|b') \right)^{\frac{a}{2}}, \quad (\text{F.7})$$

adding a non-negative scalar to the right-most expression yields

$$|r(b, u) - r(b', u)| \leq (4 \cdot \ln 2)^{\frac{a}{2}} \cdot \lambda_a \cdot \left(\frac{1}{2} \mathbb{D}_{KL}(b\|b') + \frac{1}{2} \mathbb{D}_{KL}(b'\|b) \right)^{\frac{a}{2}}, \quad (\text{F.8})$$

and finally using Eq. (E.1) we get

$$|r(b, u) - r(b', u)| \leq (4 \cdot \ln 2)^{\frac{a}{2}} \cdot \lambda_a \cdot \mathbb{D}_{\sqrt{J}}^a(b, b'). \quad (\text{F.9})$$

□

*Never put a sock
in a toaster.*

Eddie Izzard

G

Proof of Corollary 2.

Corollary 2 (of Theorem 1). *Let $r(b, u)$ be α -Hölder continuous with λ_α and $\alpha \in (0, 1]$. Let b and b' denote two beliefs. Let ω_i denote a positive weight, such that $0 \leq \omega_i \leq 1$, $i \in \{1, 2\}$. Then the weighted difference between $r(b, u)$ and $r(b', u)$ is given by*

$$(\omega_1 - \omega_2)r(b') - \omega_1\bar{\lambda}_\alpha\mathbb{D}_{\sqrt{J}}^\alpha(b, b') \leq \omega_1r(b) - \omega_2r(b') \leq \omega_1\bar{\lambda}_\alpha\mathbb{D}_{\sqrt{J}}^\alpha(b, b') + (\omega_1 - \omega_2)r(b') \quad (\text{G.1})$$

where

$$\bar{\lambda}_\alpha \triangleq \lambda_\alpha (4 \cdot \ln 2)^{\frac{\alpha}{2}}. \quad (\text{G.2})$$

Proof. Following Theorem 1, using the definition of (G.2)

$$| r(b, u) - r(b', u) | \leq \bar{\lambda}_\alpha\mathbb{D}_{\sqrt{J}}^\alpha(b, b') \quad (\text{G.3})$$

$$-\bar{\lambda}_\alpha\mathbb{D}_{\sqrt{J}}^\alpha(b, b') \leq r(b) - r(b') \leq \bar{\lambda}_\alpha\mathbb{D}_{\sqrt{J}}^\alpha(b, b') \quad (\text{G.4})$$

$$-\omega_1\bar{\lambda}_\alpha\mathbb{D}_{\sqrt{J}}^\alpha(b, b') \leq \omega_1r(b) - \omega_1r(b') \leq \omega_1\bar{\lambda}_\alpha\mathbb{D}_{\sqrt{J}}^\alpha(b, b') \quad (\text{G.5})$$

$$-\omega_1\bar{\lambda}_\alpha\mathbb{D}_{\sqrt{J}}^\alpha(b, b') \leq \omega_1r(b) - \omega_2r(b') + (\omega_2 - \omega_1)r(b') \leq \omega_1\bar{\lambda}_\alpha\mathbb{D}_{\sqrt{J}}^\alpha(b, b') \quad (\text{G.6})$$

$$(\omega_1 - \omega_2)r(b') - \omega_1\bar{\lambda}_\alpha\mathbb{D}_{\sqrt{J}}^\alpha(b, b') \leq \omega_1r(b) - \omega_2r(b') \leq \omega_1\bar{\lambda}_\alpha\mathbb{D}_{\sqrt{J}}^\alpha(b, b') + (\omega_1 - \omega_2)r(b') \quad (\text{G.7})$$

□

*People who think they know everything
are a great annoyance to those of us who do.*

Issac Asimov

H

Proof of Theorem 2.

Theorem 2. Let $r(b,u)$ be α -Hölder continuous with λ_α and $\alpha \in (0, 1]$. Let $J_{k+l|k+l}$ and $J_{k+l|k}$ be objective values of the same time step $k+l$, calculated based on information up to time $k+l$ and k respectively. Let L be a planning horizon such that $L \geq k+l+1$. Let n_i be the number of samples used to estimate the expected reward at lookahead step i . Let ω_i^j be non-negative weights such that $0 \leq \omega_i^j \leq 1$ and $\sum_{j=1}^{n_i} \omega_i^j = 1$. Then the difference $(J_{k+l|k+l} - J_{k+l|k})$ is bounded by

$$\sum_{i=k+l+1}^L \sum_{j=1}^{n_i} \omega_{i|k+l}^j [r_{i|k}^j - \mathcal{D}_i^j] - J_{k+l|k} \leq J_{k+l|k+l} - J_{k+l|k} \leq \sum_{i=k+l+1}^L \sum_{j=1}^{n_i} \omega_{i|k+l}^j [r_{i|k}^j + \mathcal{D}_i^j] - J_{k+l|k} \quad (\text{H.1})$$

where

$$\mathcal{D}_i^j = \lambda_\alpha (4 \cdot \ln 2)^{\frac{\alpha}{2}} \mathbb{D}_{\sqrt{J}}^{\alpha} (b^j[X_{i|k+l}], b^j[X_{i|k}]). \quad (\text{H.2})$$

Proof. By definition,

$$J_{k+l|k+l} - J_{k+l|k} = \sum_{i=k+l+1}^L [\mathbb{E} r_i(b[X_{i|k+l}], u) - \mathbb{E} r_i(b[X_{i|k}], u)], \quad (\text{H.3})$$

assuming the measurement likelihood is not explicitly available, we approximate the expectation using samples,

$$J_{k+l|k+l} - J_{k+l|k} \approx \sum_{i=k+l+1}^L \left[\sum_{j=1}^{n_i} \omega_{i|k+l}^j r_i^j(b^j[X_{i|k+l}], u) - \sum_{j=1}^{n_i} \omega_{i|k}^j r_i^j(b^j[X_{i|k}], u) \right] \quad (\text{H.4})$$

$$= \sum_{i=k+l+1}^L \sum_{j=1}^{n_i} \left[\omega_{i|k+l}^j r_{i|k+l}^j - \omega_{i|k}^j r_{i|k}^j \right]. \quad (\text{H.5})$$

Using Corollary 2, for specific i, j we can write

$$(\omega_{i|k+l}^j - \omega_{i|k}^j) r_{i|k}^j - \omega_{i|k+l}^j \mathcal{D}_i^j \leq \omega_{i|k+l}^j r_{i|k+l}^j - \omega_{i|k}^j r_{i|k}^j \leq \omega_{i|k+l}^j \mathcal{D}_i^j + (\omega_{i|k+l}^j - \omega_{i|k}^j) r_{i|k}^j \quad (\text{H.6})$$

where

$$\mathcal{D}_i^j = \lambda_a (4 \cdot \ln 2)^{\frac{a}{2}} \mathbb{D}_{\sqrt{J}}^a(b^j[X_{i|k+l}], b^j[X_{i|k}]). \quad (\text{H.7})$$

So following Eq. (H.6)

$$\begin{aligned} \sum_{i=k+l+1}^L \sum_{j=1}^{n_i} \left[(\omega_{i|k+l}^j - \omega_{i|k}^j) r_{i|k}^j - \omega_{i|k+l}^j \mathcal{D}_i^j \right] &\leq J_{k+l|k+l} - J_{k+l|k} \\ &\leq \sum_{i=k+l+1}^L \sum_{j=1}^{n_i} \left[(\omega_{i|k+l}^j - \omega_{i|k}^j) r_{i|k}^j + \omega_{i|k+l}^j \mathcal{D}_i^j \right] \end{aligned} \quad (\text{H.8})$$

$$\begin{aligned} \sum_{i=k+l+1}^L \sum_{j=1}^{n_i} \omega_{i|k+l}^j \left[r_{i|k}^j - \mathcal{D}_i^j \right] - J_{k+l|k} &\leq J_{k+l|k+l} - J_{k+l|k} \\ &\leq \sum_{i=k+l+1}^L \sum_{j=1}^{n_i} \omega_{i|k+l}^j \left[r_{i|k}^j + \mathcal{D}_i^j \right] - J_{k+l|k}. \end{aligned} \quad (\text{H.9})$$

□

We can simplify this further by assuming that $J_{k+l|k+l}$ is estimated using samples from the nominal measurement likelihood, so the $\omega_{i|k+l}^j$ weights are simply given by $\frac{1}{n_i} \forall j$,

$$\sum_{i=k+l+1}^L \frac{1}{n_i} \sum_{j=1}^{n_i} \left[r_{i|k}^j - \mathcal{D}_i^j \right] - J_{k+l|k} \leq J_{k+l|k+l} - J_{k+l|k} \leq \sum_{i=k+l+1}^L \frac{1}{n_i} \sum_{j=1}^{n_i} \left[r_{i|k}^j + \mathcal{D}_i^j \right] - J_{k+l|k}. \quad (\text{H.10})$$

You want proof??!
I'll give you proof!

Sidney Harris

I

Proof of Theorem 3.

Theorem 3. Let $r(b,u)$ be α -Hölder continuous with λ_α and $\alpha \in (0, 1]$. Let $J_{k+l|k+l}$ and $J_{k+l|k}$ be objective values of the same time step $k+l$, calculated based on information up to time $k+l$ and k respectively. Let L be a planning horizon such that $L \geq l+1$. Then the difference $(J_{k+l|k+l} - J_{k+l|k})$ is bounded by

$$\varphi - \psi \leq J_{k+l|k+l} - J_{k+l|k} \leq \varphi + \psi, \quad (\text{I.1})$$

where

$$\varphi \triangleq \sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} (\omega - 1) r_{i|k} \quad (\text{I.2})$$

$$\psi \triangleq \lambda_\alpha (4 \cdot \ln 2)^{\frac{\alpha}{2}} \left[(L-l) \varepsilon_{wf}^\alpha + \sum_{i=k+l+1}^{k+L} \left(\sum_{j=k+l+1}^i \mathbb{E}_{z \sim p_{k+l}} \Delta_j \right)^{\frac{\alpha}{2}} \right], \quad (\text{I.3})$$

and

$$\mathbb{D}_{\sqrt{f}}(b[X_{i|k+l}], b[X_{i|k}]) = \sqrt{\mathbb{D}_{\sqrt{f}}^2(b[X_{i-1|k+l}], b[X_{i-1|k}]) + \Delta_i}, \quad (\text{I.4})$$

$$\varepsilon_{wf} = \mathbb{D}_{\sqrt{f}}(b[X_{k+l|k+l}], b[X_{k+l|k}]), \quad (\text{I.5})$$

$$\omega = \frac{\mathbb{P}(z_{k+l+1:k+L} | H_{k+l|k+l}, u_{k+l:k+L-1})}{\mathbb{P}(z_{k+l+1:k+L} | H_{k+l|k}, u_{k+l:k+L-1})} \triangleq \frac{p_{k+l}}{p_k}. \quad (\text{I.6})$$

Proof. By definition

$$J_{k+l|k+l} - J_{k+l|k} = \sum_{i=k+l+1}^{k+L} \left[\mathbb{E}_{z_{k+l+1:k+L|k+l}} r_i(b[X_{i|k+l}], u) - \mathbb{E}_{z_{k+l+1:k+L|k}} r_i(b[X_{i|k}], u) \right], \quad (\text{I.7})$$

where each expectation is over a different measurement likelihood. In order to use a single expectation over both rewards we use importance sampling (see Eq. (D.1)). For simplicity we define ω as the likelihood ratio

$$\omega = \frac{\mathbb{P}(z_{k+l+1:k+L}|H_{k+l|k+l}, u_{k+l:k+L-1})}{\mathbb{P}(z_{k+l+1:k+L}|H_{k+l|k}, u_{k+l:k+L-1})} = \frac{p_{k+l}}{p_k}, \quad (\text{I.8})$$

and can now re-write the objective difference under a single expectation

$$J_{k+l|k+l} - J_{k+l|k} = \sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} [\omega r_{i|k+l} - r_{i|k}]. \quad (\text{I.9})$$

Following Corollary 2 we can bound the reward difference in (I.9) with

$$(\omega - 1)r_{i|k} - \omega \mathcal{D}_i \leq \omega r_{i|k+l} - r_{i|k} \leq \omega \mathcal{D}_i + (\omega - 1)r_{i|k}, \quad (\text{I.10})$$

where

$$\mathcal{D}_i = \lambda_\alpha (4 \cdot \ln 2)^{\frac{\alpha}{2}} \mathbb{D}_{\sqrt{f}}^{\alpha}(b[X_{i|k+l}], b[X_{i|k}]). \quad (\text{I.11})$$

So the objective value difference is bounded by

$$\sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} [(\omega - 1)r_{i|k} - \omega \mathcal{D}_i] \leq J_{k+l|k+l} - J_{k+l|k} \leq \sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} [\omega \mathcal{D}_i + (\omega - 1)r_{i|k}] \quad (\text{I.12})$$

$$\sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} (\omega - 1)r_{i|k} - \mathbb{E}_{z \sim p_k} \omega \mathcal{D}_i \leq J_{k+l|k+l} - J_{k+l|k} \leq \sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} \omega \mathcal{D}_i + \mathbb{E}_{z \sim p_k} (\omega - 1)r_{i|k}, \quad (\text{I.13})$$

or in a more compact manner

$$|J_{k+l|k+l} - J_{k+l|k} - \sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} (\omega - 1)r_{i|k}| \leq \sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} \omega \mathcal{D}_i. \quad (\text{I.14})$$

Let us define the delta distance between two consecutive lookahead steps as,

$$\mathbb{D}_{\sqrt{f}}(b[X_{i|k+l}], b[X_{i|k}]) = \sqrt{\mathbb{D}_{\sqrt{f}}^2(b[X_{i-1|k+l}], b[X_{i-1|k}]) + \Delta_i}, \quad (\text{I.15})$$

so two sequential yet non consecutive lookahead steps can be written as

$$\mathbb{D}_{\sqrt{J}}(b[X_{i+1|k+l}], b[X_{i+1|k}]) = \sqrt{\mathbb{D}_{\sqrt{J}}^2(b[X_{i-1|k+l}], b[X_{i-1|k}]) + \Delta_i + \Delta_{i+1}}, \quad (\text{I.16})$$

We will now continue with simplifying the RHS of (I.14),

$$\sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} \omega \mathcal{D}_i = \sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} \omega \lambda_a (4 \cdot \ln 2)^{\frac{a}{2}} \mathbb{D}_{\sqrt{J}}^a(b[X_{i|k+l}], b[X_{i|k}]) \quad (\text{I.17})$$

$$= \lambda_a (4 \cdot \ln 2)^{\frac{a}{2}} \sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} \omega \mathbb{D}_{\sqrt{J}}^a(b[X_{i|k+l}], b[X_{i|k}]), \quad (\text{I.18})$$

using the connection given by (I.16) as well as moving back to expectation over p_{k+l} we get

$$= \lambda_a (4 \cdot \ln 2)^{\frac{a}{2}} \sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_{k+l}} \left[\mathbb{D}_{\sqrt{J}}^2(b[X_{k+l|k+l}], b[X_{k+l|k}]) + \sum_{j=k+l+1}^i \Delta_j \right]^{\frac{a}{2}}, \quad (\text{I.19})$$

$$\leq \lambda_a (4 \cdot \ln 2)^{\frac{a}{2}} \sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_{k+l}} \left[\varepsilon_{wf}^2 + \sum_{j=k+l+1}^i \Delta_j \right]^{\frac{a}{2}}, \quad (\text{I.20})$$

where $\frac{a}{2} \in (0, \frac{1}{2}]$ so $(.)^{\frac{a}{2}}$ is a concave function, and $-(.)^{\frac{a}{2}}$ is convex and following Jensen inequality we get

$$\sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} \omega \mathcal{D}_i \leq \lambda_a (4 \cdot \ln 2)^{\frac{a}{2}} \left[(L-l) \varepsilon_{wf}^a + \sum_{i=k+l+1}^{k+L} \left(\sum_{j=k+l+1}^i \mathbb{E}_{z \sim p_{k+l}} \Delta_j \right)^{\frac{a}{2}} \right]. \quad (\text{I.21})$$

Going back to (I.14),

$$\begin{aligned} |J_{k+l|k+l} - J_{k+l|k} - \sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} (\omega - 1) r_{i|k}| &\leq \sum_{i=k+l+1}^{k+L} \mathbb{E}_{z \sim p_k} \omega \mathcal{D}_i \\ &\leq \lambda_a (4 \cdot \ln 2)^{\frac{a}{2}} \left[(L-l) \varepsilon_{wf}^a + \sum_{i=k+l+1}^{k+L} \left(\sum_{j=k+l+1}^i \mathbb{E}_{z \sim p_{k+l}} \Delta_j \right)^{\frac{a}{2}} \right]. \end{aligned} \quad (\text{I.22})$$

Using the definitions of φ and ψ given respectively by Eq.(I.2) and Eq.(I.3), we can reformulate (I.22) into

$$\varphi - \psi \leq J_{k+l|k+l} - J_{k+l|k} \leq \varphi + \psi. \quad (\text{I.23})$$

□

For the special case where $\omega = 1$ we get

$$|J_{k+l|k+l} - J_{k+l|k}| \leq (4 \cdot \ln 2)^{\frac{a}{2}} \cdot \lambda_a \cdot \left[(L - l) \cdot \varepsilon_{wf}^a + \sum_{i=k+l+1}^{k+L} \left(\sum_{j=k+l+1}^i \mathbb{E} \Delta_j \right)^{\frac{a}{2}} \right]. \quad (\text{I.24})$$

*Prejudice is a great time saver.
You can form opinions without having to get the facts.*

E. B. White

J

Proof of Lemma 2

Lemma 2 (Incremental \mathbb{D}_{PQ} distance). Let b_1 and b_2 be two gaussian beliefs with (μ_1, Σ_1) and (μ_2, Σ_2) respectively, and their two (differently) propagated counterparts b_{1p} and b_{2p} with (μ_{1p}, Σ_{1p}) and (μ_2^+, Σ_{2p}) . When the propagated mean and covariance are defined as

$$\mu_{ip} = \mu_i + \zeta_i \quad , \quad \Sigma_{ip} = (\Sigma_i^{-1} + A_i^T A_i)^{-1} \quad , \quad i \in [1, 2] \quad (\text{J.1})$$

Then the squared \mathbb{D}_{PQ} distance between the propagated beliefs can be written as

$$\mathbb{D}_{\sqrt{J}}^2(b_{1p}, b_{2p}) = \mathbb{D}_{\sqrt{J}}^2(b_1, b_2) + \Delta, \quad (\text{J.2})$$

where

$$\begin{aligned} \Delta &= \frac{1}{4}(\mu_2 - \mu_1)^T [A_2^T A_2 + A_1^T A_1] (\mu_2 - \mu_1) + \frac{1}{2}(\mu_2 - \mu_1)^T [\Sigma_{1p}^{-1} + \Sigma_{2p}^{-1}] (\zeta_2 - \zeta_1) \\ &\quad + \frac{1}{4}(\zeta_2 - \zeta_1)^T [\Sigma_{1p}^{-1} + \Sigma_{2p}^{-1}] (\zeta_2 - \zeta_1) + \frac{1}{4} \operatorname{tr} (A_2^T A_2 \Sigma_{1p} - \Sigma_2^{-1} \Sigma_1 A_1^T (I + A_1 \Sigma_1 A_1^T)^{-1} A_1 \Sigma_1) \\ &\quad + \frac{1}{4} \operatorname{tr} (A_1^T A_1 \Sigma_{2p} - \Sigma_1^{-1} \Sigma_2 A_2^T (I + A_2 \Sigma_2 A_2^T)^{-1} A_2 \Sigma_2) - \frac{1}{2} (d_p - d). \quad (\text{J.3}) \end{aligned}$$

Proof. The \mathbb{D}_{PQ} distance between b_1 and b_2 is thus given by

$$\begin{aligned} \mathbb{D}_{\sqrt{J}}^2(b_1, b_2) &= \\ &\quad \frac{1}{4} [(\mu_2 - \mu_1)^T [\Sigma_1^{-1} + \Sigma_2^{-1}] (\mu_2 - \mu_1) + \operatorname{tr}(\Sigma_2^{-1} \Sigma_1) + \operatorname{tr}(\Sigma_1^{-1} \Sigma_2) - d_1 - d_2], \quad (\text{J.4}) \end{aligned}$$

when d_i represents the dimension of the un-zero-padded Σ_i . Equivalently the distance between b_{1p}

and b_{2p} is given by

$$\begin{aligned} \mathbb{D}_{\sqrt{J}}^2(b_{1p}, b_{2p}) = & \\ \frac{1}{4} \left[(\mu_{2p} - \mu_{1p})^T [\Sigma_{1p}^{-1} + \Sigma_{2p}^{-1}] (\mu_{2p} - \mu_{1p}) + \text{tr}(\Sigma_{2p}^{-1} \Sigma_{1p}) + \text{tr}(\Sigma_{1p}^{-1} \Sigma_{2p}) - d_{1p} - d_{2p} \right] \quad (\text{J.5}) \end{aligned}$$

where d_i^+ represents the dimension of the un-zero-padded Σ_i^+ .

We would like to show that

$$\mathbb{D}_{\sqrt{J}}^2(b_{1p}, b_{2p}) = \mathbb{D}_{\sqrt{J}}^2(b_1, b_2) + \Delta. \quad (\text{J.6})$$

We start from substituting Eq. (J.1) in the trace expression from Eq. (J.5)

$$\text{tr}(\Sigma_{2p}^{-1} \Sigma_{1p}) = \text{tr}((\Sigma_2^{-1} + A_2^T A_2) (\Sigma_1^{-1} + A_1^T A_1)^{-1})$$

using Woodbury matrix identity [75] we can rewrite it as

$$\text{tr}(\Sigma_{2p}^{-1} \Sigma_{1p}) = \text{tr}((\Sigma_2^{-1} + A_2^T A_2) (\Sigma_1 - \Sigma_1 A_1^T (I + A_1 \Sigma_1 A_1^T)^{-1} A_1 \Sigma_1)).$$

After some simple manipulations we get

$$\text{tr}(\Sigma_{2p}^{-1} \Sigma_{1p}) = \text{tr}(\Sigma_2^{-1} \Sigma_1) + \text{tr}(A_2^T A_2 \Sigma_{1p} - \Sigma_2^{-1} \Sigma_1 A_1^T (I + A_1 \Sigma_1 A_1^T)^{-1} A_1 \Sigma_1). \quad (\text{J.7})$$

In a similar manner we can get an expression for the symmetric trace expression in Eq. (J.5)

$$\text{tr}(\Sigma_{1p}^{-1} \Sigma_{2p}) = \text{tr}(\Sigma_1^{-1} \Sigma_2) + \text{tr}(A_1^T A_1 \Sigma_{2p} - \Sigma_1^{-1} \Sigma_2 A_2^T (I + A_2 \Sigma_2 A_2^T)^{-1} A_2 \Sigma_2). \quad (\text{J.8})$$

We are left with the square root Mahalanobis distance expression in Eq. (J.5). By substituting Eq. (J.1) in the aforementioned we get

$$\begin{aligned} & (\mu_{2p} - \mu_{1p})^T [\Sigma_{1p}^{-1} + \Sigma_{2p}^{-1}] (\mu_{2p} - \mu_{1p}) \\ &= (\mu_2 + \zeta_2 - \mu_1 - \zeta_1)^T [\Sigma_1^{-1} + A_1^T A_1 + \Sigma_2^{-1} + A_2^T A_2] (\mu_2 + \zeta_2 - \mu_1 - \zeta_1) \\ &= (\mu_2 - \mu_1)^T [\Sigma_1^{-1} + \Sigma_2^{-1}] (\mu_2 - \mu_1) + (\mu_2 - \mu_1)^T [A_2^T A_2 + A_1^T A_1] (\mu_2 - \mu_1) + \\ & 2(\mu_2 - \mu_1)^T [\Sigma_{1p}^{-1} + \Sigma_{2p}^{-1}] (\zeta_2 - \zeta_1) + (\zeta_2 - \zeta_1)^T [\Sigma_{1p}^{-1} + \Sigma_{2p}^{-1}] (\zeta_2 - \zeta_1). \quad (\text{J.9}) \end{aligned}$$

By substituting Eqs. (J.7)-(J.9) in Eq. (J.5) we receive

$$\mathbb{D}_{\sqrt{J}}^2(b_{1p}, b_{2p}) = \mathbb{D}_{\sqrt{J}}^2(b_1, b_2) + \Delta, \quad (\text{J.10})$$

where

$$\begin{aligned}\Delta &= \frac{1}{4}(\mu_2 - \mu_1)^T [A_2^T A_2 + A_1^T A_1] (\mu_2 - \mu_1) + \frac{1}{2}(\mu_2 - \mu_1)^T [\Sigma_{1p}^{-1} + \Sigma_{2p}^{-1}] (\zeta_2 - \zeta_1) \\ &+ \frac{1}{4}(\zeta_2 - \zeta_1)^T [\Sigma_{1p}^{-1} + \Sigma_{2p}^{-1}] (\zeta_2 - \zeta_1) + \frac{1}{4} \text{tr} (A_2^T A_2 \Sigma_{1p} - \Sigma_2^{-1} \Sigma_1 A_1^T (I + A_1 \Sigma_1 A_1^T)^{-1} A_1 \Sigma_1) \\ &+ \frac{1}{4} \text{tr} (A_1^T A_1 \Sigma_{2p} - \Sigma_1^{-1} \Sigma_2 A_2^T (I + A_2 \Sigma_2 A_2^T)^{-1} A_2 \Sigma_2) - \frac{1}{2} (d_p - d). \quad (\text{J.11})\end{aligned}$$

□

When considering future beliefs, which are a function of future measurements, the belief solution is a random variable depending on the future measurement. As such Δ in Eq. (J.11), which is a function of belief solution, is in-fact a random variable.

I'm sorry, if you were right,
I'd agree with you.

Robin Williams

K

Proof of Lemma 3.

Lemma 5 (state estimation increment as a random variable). *Let μ denote the state estimate of some gaussian belief $b = \mathcal{N}(\mu_o, \Sigma_o)$. Let b_p denote the gaussian belief resulting from propagating b with some action u and some measurements z using the linear Gaussian models (6.49)-(6.50). Let μ_p denote the state estimate b_p and define a new random variable $\zeta = \mu_p - \mu$. Then*

$$\zeta \sim \mathcal{N}(\mu_\zeta, \Sigma_\zeta), \quad (\text{K.1})$$

where

$$\mu_\zeta = \sigma_{21} \left(\Sigma_o^{-1} \mu_o - \mathcal{F}^T \Sigma_w^{-1} \mathcal{J} u \right) + \sigma_{22} \left(\Sigma_w^{-1} \mathcal{J} u + \mathcal{H}^T \Sigma_v^{-1} (\mathcal{H} \mathcal{F} \mu_o + \mathcal{H} \mathcal{J} u) \right) - \mu_o \quad (\text{K.2})$$

$$\Sigma_\zeta = \sigma_{22} \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \mathcal{F} \Sigma_o \mathcal{F}^T \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \sigma_{22} + \sigma_{22} \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \Sigma_w \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \sigma_{22} + \sigma_{22} \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \sigma_{22} \quad (\text{K.3})$$

$$\begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix} = \begin{bmatrix} \Sigma_o^{-1} + \mathcal{F}^T \Sigma_w^{-1} \mathcal{F} & -\mathcal{F}^T \Sigma_w^{-1} \\ -\Sigma_w^{-1} \mathcal{F} & \Sigma_w^{-1} + \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \end{bmatrix}^{-1}. \quad (\text{K.4})$$

Proof. see Appendix L. □

Lemma 3 (Incremental $\mathbb{D}_{\sqrt{f}}$ distance as Gaussian Quadratic). *Let b_1 and b_2 be two Gaussian beliefs $\mathcal{N}(\mu_1, \Sigma_1)$ and $\mathcal{N}(\mu_2, \Sigma_2)$, respectively with state dimension d , and their two propagated counterparts b_{1p} and b_{2p} with $\mathcal{N}(\mu_{1p}, \Sigma_{1p})$ $\mathcal{N}(\mu_{2p}, \Sigma_{2p})$ and with state dimension d_p . There exist ζ_i and A_i such that the propagated mean and covariance are given by,*

$$\mu_{ip} = \mu_i + \zeta_i \quad , \quad \Sigma_{ip} = (\Sigma_i^{-1} + A_i^T A_i)^{-1} \quad , \quad i \in [1, 2]. \quad (\text{K.5})$$

Then the incremental $\mathbb{D}_{\sqrt{J}}$ distance $\Delta \triangleq \mathbb{D}_{\sqrt{J}}^2(b_{1p}, b_{2p}) - \mathbb{D}_{\sqrt{J}}^2(b_1, b_2)$ is a quadratic form of a gaussian variable.

Proof. The incremental $\mathbb{D}_{\sqrt{J}}$ distance under the assumption of linear Gaussian models is given by Lemma 2. Let us define a new variable \mathcal{S} such that

$$\mathcal{S} = \zeta_2 - \zeta_1. \quad (\text{K.6})$$

We can now re-write the result for Δ given by Lemma 2 in terms of \mathcal{S} ,

$$\Delta(\mathcal{S}) = \mathcal{S}^T C \mathcal{S} + c^T \mathcal{S} + y, \quad (\text{K.7})$$

where

$$C = \frac{1}{4} [\Sigma_{1p}^{-1} + \Sigma_{2p}^{-1}] \quad (\text{K.8})$$

$$c = \frac{1}{2} (\mu_2 - \mu_1)^T [\Sigma_{1p}^{-1} + \Sigma_{2p}^{-1}] \quad (\text{K.9})$$

$$\begin{aligned} y &= \frac{1}{4} (\mu_2 - \mu_1)^T [A_2^T A_2 + A_1^T A_1] (\mu_2 - \mu_1) \\ &\quad + \frac{1}{4} \text{tr} (A_2^T A_2 \Sigma_{1p} - \Sigma_2^{-1} \Sigma_1 A_1^T (I + A_1 \Sigma_1 A_1^T)^{-1} A_1 \Sigma_1) \\ &\quad + \frac{1}{4} \text{tr} (A_1^T A_1 \Sigma_{2p} - \Sigma_1^{-1} \Sigma_2 A_2^T (I + A_2 \Sigma_2 A_2^T)^{-1} A_2 \Sigma_2) - \frac{1}{2} (d_p - d). \end{aligned} \quad (\text{K.10})$$

Eq (K.7) presents Δ as a quadratic form of \mathcal{S} leaving us to find how \mathcal{S} is distributed.

Following Lemma 5, we know the distribution of ζ_i , thus from being a linear combination of Gaussian variables we know the distribution of \mathcal{S} to be also Gaussian

$$\mathcal{S} \sim \mathcal{N}(\mu_{\mathcal{S}}, \Sigma_{\mathcal{S}}) \quad (\text{K.11})$$

where

$$\mu_{\mathcal{S}} = \mu_{\zeta_2} - \mu_{\zeta_1}, \quad (\text{K.12})$$

$$\Sigma_{\mathcal{S}} = \Sigma_{\zeta_2} + \Sigma_{\zeta_1} + 2\Sigma_{\zeta_1 \zeta_2}. \quad (\text{K.13})$$

So Δ is a quadratic form of the Gaussian variable \mathcal{S} . □

*When I eventually met Mr. Right
I had no idea that his first name was Always.*

Rita Rudner

L

Proof of Lemma 5.

Lemma 5 (state estimation increment as a random variable). *Let μ denote the state estimate of some gaussian belief $b = \mathcal{N}(\mu_o, \Sigma_o)$. Let b_p denote the gaussian belief resulting from propagating b with some action u and some measurements z using the linear Gaussian models (6.49)-(6.50). Let μ_p denote the state estimate b_p and define a new random variable $\zeta = \mu_p - \mu$. Then*

$$\zeta \sim \mathcal{N}(\mu_\zeta, \Sigma_\zeta), \quad (\text{L.1})$$

where

$$\mu_\zeta = \sigma_{21} (\Sigma_o^{-1} \mu_o - \mathcal{F}^T \Sigma_w^{-1} \mathcal{J} u) + \sigma_{22} (\Sigma_w^{-1} \mathcal{J} u + \mathcal{H}^T \Sigma_v^{-1} (\mathcal{H} \mathcal{F} \mu_o + \mathcal{H} \mathcal{J} u)) - \mu_o \quad (\text{L.2})$$

$$\Sigma_\zeta = \sigma_{22} \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \mathcal{F} \Sigma_o \mathcal{F}^T \mathcal{H} \Sigma_v^{-1} \mathcal{H} \sigma_{22} + \sigma_{22} \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \Sigma_w \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \sigma_{22} + \sigma_{22} \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \sigma_{22} \quad (\text{L.3})$$

$$\begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix} = \begin{bmatrix} \Sigma_o^{-1} + \mathcal{F}^T \Sigma_w^{-1} \mathcal{F} & -\mathcal{F}^T \Sigma_w^{-1} \\ -\Sigma_w^{-1} \mathcal{F} & \Sigma_w^{-1} + \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \end{bmatrix}^{-1}. \quad (\text{L.4})$$

Proof. Because the belief b_p resulted from propagating the belief b with motion and measurements, using Bayes rule we can write it as proportional to

$$b_p \propto b \cdot \mathbb{P}(x'|x, u) \mathbb{P}(z|x'), \quad (\text{L.5})$$

without affecting generality let us assume a single measurement is considered. Denoting $X = \begin{bmatrix} x \\ x' \end{bmatrix}$,

the inference solution of b_p can be obtained through MAP estimation

$$\hat{X} = \arg \max_X b \cdot \mathbb{P}(x'|x, u) \mathbb{P}(z|x'), \quad (\text{L.6})$$

taking the negative log yields the following LS problem

$$\hat{X} = \arg \min_X \|x - \mu_o\|_{\Sigma_o}^2 + \|x' - \mathcal{F}x - \mathcal{J}u\|_{\Sigma_w}^2 + \|z - \mathcal{H}x\|_{\Sigma_v}^2, \quad (\text{L.7})$$

where (6.49)-(6.50) were used for linear motion and measurement models with zero mean Gaussian noises

$$\begin{aligned} x' &= \mathcal{F}x + \mathcal{J}u + w \quad , \quad w \sim \mathcal{N}(0, \Sigma_w), \\ z &= \mathcal{H}x + v \quad , \quad v \sim \mathcal{N}(0, \Sigma_v). \end{aligned}$$

We can further reformulate the problem into a LS form

$$\hat{X} = \arg \min_X \|A \cdot X - b\|^2, \quad (\text{L.8})$$

where

$$A = \begin{bmatrix} \Sigma_o^{-\frac{1}{2}} & 0 \\ -\Sigma_w^{-\frac{1}{2}} \mathcal{F} & \Sigma_w^{-\frac{1}{2}} \\ 0 & \Sigma_v^{-\frac{1}{2}} \mathcal{H} \end{bmatrix}, \quad b = \begin{bmatrix} \Sigma_o^{-\frac{1}{2}} \mu_o \\ \Sigma_w^{-\frac{1}{2}} \mathcal{J}u \\ \Sigma_v^{-\frac{1}{2}} z \end{bmatrix}. \quad (\text{L.9})$$

The solution to the inference problem is given by

$$X \sim \mathcal{N}(\Sigma A^T b, \Sigma) \quad (\text{L.10})$$

where the joint covariance matrix is given by

$$\Sigma_j = (A^T A)^{-1} = \left(\begin{bmatrix} \Sigma_o^{-\frac{1}{2}} & -\mathcal{F}^T \Sigma_w^{-\frac{1}{2}} & 0 \\ 0 & \Sigma_w^{-\frac{1}{2}} & \mathcal{H}^T \Sigma_v^{-\frac{1}{2}} \end{bmatrix} \cdot \begin{bmatrix} \Sigma_o^{-\frac{1}{2}} & 0 \\ -\Sigma_w^{-\frac{1}{2}} \mathcal{F} & \Sigma_w^{-\frac{1}{2}} \\ 0 & \Sigma_v^{-\frac{1}{2}} \mathcal{H} \end{bmatrix} \right)^{-1} \quad (\text{L.11})$$

$$\Sigma_j = \begin{bmatrix} \Sigma_o^{-1} + \mathcal{F}^T \Sigma_w^{-1} \mathcal{F} & -\mathcal{F}^T \Sigma_w^{-1} \\ -\Sigma_w^{-1} \mathcal{F} & \Sigma_w^{-1} + \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \end{bmatrix}^{-1} = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix}, \quad (\text{L.12})$$

for the readers convenience we denote the block matrices of Σ_j by $\sigma_{i,j}$, where they can be calculated explicitly through the Schur complement. So the joint state estimation is given by

$$\hat{X} = \begin{bmatrix} \sigma_{11} & \sigma_{12} \\ \sigma_{21} & \sigma_{22} \end{bmatrix} \begin{bmatrix} \Sigma_o^{-\frac{1}{2}} & -\mathcal{F}^T \Sigma_w^{-\frac{1}{2}} & 0 \\ 0 & \Sigma_w^{-\frac{1}{2}} & \mathcal{H}^T \Sigma_v^{-\frac{1}{2}} \end{bmatrix} \begin{bmatrix} \Sigma_o^{-\frac{1}{2}} \mu_o \\ \Sigma_w^{-\frac{1}{2}} \mathcal{J}u \\ \Sigma_v^{-\frac{1}{2}} z \end{bmatrix}, \quad (\text{L.13})$$

and we can write the state estimation of x' explicitly as

$$\mu_p = \hat{x}' = \sigma_{21} (\Sigma_o^{-1} \mu_o - \mathcal{F}^T \Sigma_w^{-1} \mathcal{J} u) + \sigma_{22} (\Sigma_w^{-1} \mathcal{J} u + \mathcal{H}^T \Sigma_v^{-1} z). \quad (\text{L.14})$$

Now that we have developed the state estimation, we are in position to express ζ in terms of the state estimations

$$\zeta = \mu_p - \mu_o = \sigma_{21} (\Sigma_o^{-1} \mu_o - \mathcal{F}^T \Sigma_w^{-1} \mathcal{J} u) + \sigma_{22} (\Sigma_w^{-1} \mathcal{J} u + \mathcal{H}^T \Sigma_v^{-1} z) - \mu_o \quad (\text{L.15})$$

through the measurement and motion models we know

$$z = \mathcal{H} \mathcal{F} x + \mathcal{H} \mathcal{J} u + \mathcal{H} w + v. \quad (\text{L.16})$$

Following z is a linear transformation of a gaussian variable, it is also a gaussian variable with

$$z \sim \mathcal{N}(\mathcal{H} \mathcal{F} \mu_o + \mathcal{H} \mathcal{J} u, \mathcal{H} \mathcal{F} \Sigma_o \mathcal{F}^T \mathcal{H}^T + \mathcal{H} \Sigma_w \mathcal{H}^T + \Sigma_v), \quad (\text{L.17})$$

the same result can be obtained by explicitly calculating the pdf of the measurement likelihood function as presented in Appendix N. Using (L.17) and the connection provided in (L.15) we can say that as a linear transformation of a gaussian variable, ζ too is a gaussian variable with

$$\zeta \sim \mathcal{N}(\mu_\zeta, \Sigma_\zeta) \quad (\text{L.18})$$

where

$$\mu_\zeta = \sigma_{21} (\Sigma_o^{-1} \mu_o - \mathcal{F}^T \Sigma_w^{-1} \mathcal{J} u) + \sigma_{22} (\Sigma_w^{-1} \mathcal{J} u + \mathcal{H}^T \Sigma_v^{-1} (\mathcal{H} \mathcal{F} \mu_o + \mathcal{H} \mathcal{J} u)) - \mu_o \quad (\text{L.19})$$

$$\Sigma_\zeta = \sigma_{22} \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \mathcal{F} \Sigma_o \mathcal{F}^T \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \sigma_{22} + \sigma_{22} \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \Sigma_w \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \sigma_{22} + \sigma_{22} \mathcal{H}^T \Sigma_v^{-1} \mathcal{H} \sigma_{22} \quad (\text{L.20})$$

□

*What's right is what's left
if you do everything else wrong.*

Robin Williams

M

Proof of Corollary 3.

Lemma 6 (Moments of Gaussian Quadratic). *Let Δ be quadratic expression $\Delta = \mathcal{S}^T C \mathcal{S} + c^T \mathcal{S} + y$, where $C^T = C$, $\mathcal{S} \sim \mathcal{N}(\mu, \Sigma)$, and $\Sigma > 0$. Then the first two moments of Δ are given by*

$$\mathbb{E}[\Delta] = \text{tr}(\Sigma^{\frac{1}{2}} C \Sigma^{\frac{1}{2}}) + \mu^T C \mu + c^T \mu + y \quad (\text{M.1})$$

$$\mathbb{E}[\Delta]^2 = 2\text{tr}(\Sigma^{\frac{1}{2}} C \Sigma C \Sigma^{\frac{1}{2}}) + \|c + 2C\mu\|_{\Sigma}^2 \quad (\text{M.2})$$

where $\|\cdot\|_{\Sigma}^2$ is the Mahalanobis distance.

Proof. This is Theorem 3.2b.3 in [52]. □

Interestingly enough, there are specified conditions under which a Gaussian quadratic form is distributed as a non-central χ^2 distribution, although not required for our proof we supply these conditions in Appendix O for completion.

Corollary 3 (of Theorem 3). *Let $r(b,u)$ be α -Hölder continuous with λ_α and $\alpha \in (0, 1]$. Let $J_{k+l|k+l}$ and $J_{k+l|k}$ be objective values of the same time step $k+l$, calculated based on information up to time $k+l$ and k respectively. Let L be a planning horizon such that $L \geq l+1$. Let the motion and measurement models be linear with additive Gaussian noise (6.49)-(6.50). Then the bound of $(J_{k+l|k+l} - J_{k+l|k})$ can be explicitly calculated.*

Proof. Following Theorem 3, the difference $(J_{k+l|k+l} - J_{k+l|k})$ is bounded where the bound is given by Eq. (I.1). In order to explicitly calculate (I.1) we are left with calculating the expected value of $\Delta_j \forall j$, where the rewards r_{i_k} are given from previously calculated planning session, and just need to be re-weighted using $(\omega - 1)$. Following Lemma 3 we know Δ_j to be a quadratic expression of the Gaussian multinomial variable $\mathcal{S}_j \sim \mathcal{N}(\mu_{\mathcal{S}_j}, \Sigma_{\mathcal{S}_j})$

$$\Delta_j = \mathcal{S}_j^T C_j \mathcal{S}_j + c_j^T \mathcal{S}_j + y_j,$$

where C_j , c_j and y_j are respectively given by Eqs. (K.8)-(K.10). As such following Lemma 6, the first moment of Δ_j is readily given by

$$\mathbb{E} \Delta_j = \text{tr}(\Sigma_{\mathcal{S}_j}^{\frac{1}{2}} C \Sigma_{\mathcal{S}_j}^{\frac{1}{2}}) + \mu_{\mathcal{S}_j}^T C_j \mu_{\mathcal{S}_j} + c_j^T \mu_{\mathcal{S}_j} + y_j. \quad (\text{M.3})$$

□

*A child of five would understand this.
Send someone to fetch me a child of five.*

Groucho Marx

N

Measurement Likelihood Under Linear Gaussian Models.

In this section we calculate the measurement likelihood probability under the assumption of linear gaussian models. We will show that under these assumptions the measurement likelihood is gaussian. Our proof starts with introducing the state into the measurement likelihood, followed by some manipulations to marginalize the state out and be left with a gaussian function over the measurement.

We start by introducing the state into the measurement likelihood, which allows us to get an equivalent expression with the measurement and motion models,

$$\mathbb{P}(z|H^-) = \int \mathbb{P}(z|x)\mathbb{P}(x|H^-)dx \quad (\text{N.1})$$

where

$$\mathbb{P}(z|x) = \mathcal{N}(Hx, \Sigma_v) \quad , \quad \mathbb{P}(x|H^-) = \mathcal{N}(\mu_p, \Sigma_p) \quad (\text{N.2})$$

$$\mu_p = F\mu_o + Ju \quad , \quad \Sigma_p = \Sigma_w + F\Sigma_o F^T. \quad (\text{N.3})$$

$$\mathbb{P}(z|H^-) = \int \frac{1}{\sqrt{(2\pi)^{d_z} |\Sigma_v|}} e^{-\frac{1}{2}\|z-Hx\|_{\Sigma_v}^2} \frac{1}{\sqrt{(2\pi)^{d_x} |\Sigma_p|}} e^{-\frac{1}{2}\|x-\mu_p\|_{\Sigma_p}^2} dx. \quad (\text{N.4})$$

Let us look only at the argument of the exponent (divided by $-\frac{1}{2}$)

$$\|z - Hx\|_{\Sigma_v}^2 + \|x - \mu_p\|_{\Sigma_p}^2 = (z - Hx)^T \Sigma_v^{-1} (z - Hx) + (x - \mu_p)^T \Sigma_p^{-1} (x - \mu_p) \quad (\text{N.5})$$

$$= z^T \Sigma_v^{-1} z - 2x^T H^T \Sigma_v^{-1} z + x^T H^T \Sigma_v^{-1} Hx + x^T \Sigma_p^{-1} x - 2x^T \Sigma_p^{-1} \mu_p + \mu_p^T \Sigma_p^{-1} \mu_p \quad (\text{N.6})$$

$$= \mathbf{x}^T (H^T \Sigma_v^{-1} H + \Sigma_p^{-1}) \mathbf{x} - 2\mathbf{x}^T (H^T \Sigma_v^{-1} z + \Sigma_p^{-1} \mu_p) + z^T \Sigma_v^{-1} z + \mu_p^T \Sigma_p^{-1} \mu_p. \quad (\text{N.7})$$

Let us define

$$\Sigma_x \doteq (H^T \Sigma_v^{-1} H + \Sigma_p^{-1})^{-1} = \Sigma_p - \Sigma_p H^T (\Sigma_v + H \Sigma_p H^T)^{-1} H \Sigma_p \quad (\text{N.8})$$

$$\mu_x \doteq (H^T \Sigma_v^{-1} H + \Sigma_p^{-1})^{-1} (H^T \Sigma_v^{-1} z + \Sigma_p^{-1} \mu_p) = \Sigma_x (H^T \Sigma_v^{-1} z + \Sigma_p^{-1} \mu_p) \quad (\text{N.9})$$

now we can reformulate (N.7) as

$$\|z - Hx\|_{\Sigma_v}^2 + \|\mathbf{x} - \mu_p\|_{\Sigma_p}^2$$

$$= \mathbf{x}^T \Sigma_x^{-1} \mathbf{x} - 2\mathbf{x}^T \Sigma_x^{-1} \mu_x + \mu_x^T \Sigma_x^{-1} \mu_x - \mu_x^T \Sigma_x^{-1} \mu_x + z^T \Sigma_v^{-1} z + \mu_p^T \Sigma_p^{-1} \mu_p \quad (\text{N.10})$$

$$= \|\mathbf{x} - \mu_x\|_{\Sigma_x}^2 - \mu_x^T \Sigma_x^{-1} \mu_x + z^T \Sigma_v^{-1} z + \mu_p^T \Sigma_p^{-1} \mu_p. \quad (\text{N.11})$$

Going back to (N.4) we get

$$\begin{aligned} & \mathbb{P}(z | H^-) \\ &= \sqrt{\frac{|\Sigma_x|}{(2\pi)^{d_z} |\Sigma_p| |\Sigma_v|}} e^{-\frac{1}{2}(z^T \Sigma_v^{-1} z + \mu_p^T \Sigma_p^{-1} \mu_p - \mu_x^T \Sigma_x^{-1} \mu_x)} \int \frac{1}{\sqrt{(2\pi)^{d_x} |\Sigma_x|}} e^{-\frac{1}{2}\|\mathbf{x} - \mu_x\|_{\Sigma_x}^2} d\mathbf{x} \quad (\text{N.12}) \end{aligned}$$

$$= \sqrt{\frac{|\Sigma_x|}{(2\pi)^{d_z} |\Sigma_p| |\Sigma_v|}} e^{-\frac{1}{2}(z^T \Sigma_v^{-1} z + \mu_p^T \Sigma_p^{-1} \mu_p - \mu_x^T \Sigma_x^{-1} \mu_x)}. \quad (\text{N.13})$$

We now try to reformulate the exponent argument in (N.13) into a quadratic form

$$z^T \Sigma_v^{-1} z + \mu_p^T \Sigma_p^{-1} \mu_p - \mu_x^T \Sigma_x^{-1} \mu_x \quad (\text{N.14})$$

$$= z^T \Sigma_v^{-1} z + \mu_p^T \Sigma_p^{-1} \mu_p - (H^T \Sigma_v^{-1} z + \Sigma_p^{-1} \mu_p)^T \Sigma_x (H^T \Sigma_v^{-1} z + \Sigma_p^{-1} \mu_p) \quad (\text{N.15})$$

$$= z^T [\Sigma_v^{-1} - \Sigma_v^{-1} H \Sigma_x H^T \Sigma_v^{-1}] z - 2z^T \Sigma_v^{-1} H \Sigma_x \Sigma_p^{-1} \mu_p - \mu_p^T \Sigma_p^{-1} \Sigma_x \Sigma_p^{-1} \mu_p + \mu_p^T \Sigma_p^{-1} \mu_p \quad (\text{N.16})$$

Let us define

$$\Sigma_z \doteq [\Sigma_v^{-1} - \Sigma_v^{-1} H \Sigma_x H^T \Sigma_v^{-1}]^{-1} = [\Sigma_v^{-1} - \Sigma_v^{-1} H (H^T \Sigma_v^{-1} H + \Sigma_p^{-1})^{-1} H^T \Sigma_v^{-1}]^{-1} \quad (\text{N.17})$$

$$\Sigma_z \doteq \Sigma_v + H \Sigma_p H^T \quad (\text{N.18})$$

$$\mu_z \doteq (\Sigma_v + H \Sigma_p H^T) \Sigma_v^{-1} H \Sigma_x \Sigma_p^{-1} \mu_p \quad (\text{N.19})$$

now we can reformulate (N.16) as

$$= z^T \Sigma_z^{-1} z - 2z^T \Sigma_z^{-1} \mu_z + \mu_z^T \Sigma_z^{-1} \mu_z - \mu_z^T \Sigma_z^{-1} \mu_z - \mu_p^T \Sigma_p^{-1} \Sigma_x \Sigma_p^{-1} \mu_p + \mu_p^T \Sigma_p^{-1} \mu_p \quad (\text{N.20})$$

$$= \|z - \mu_z\|_{\Sigma_z}^2 - \mu_z^T \Sigma_z^{-1} \mu_z - \mu_p^T \Sigma_p^{-1} \Sigma_x \Sigma_p^{-1} \mu_p + \mu_p^T \Sigma_p^{-1} \mu_p. \quad (\text{N.21})$$

Using (N.21) in (N.13) we get

$$\mathbb{P}(z|H^-) = \sqrt{\frac{|\Sigma_x||\Sigma_z|}{|\Sigma_p||\Sigma_v|}} e^{-\frac{1}{2}(-\mu_z^T \Sigma_z^{-1} \mu_z - \mu_p^T \Sigma_p^{-1} \Sigma_x \Sigma_p^{-1} \mu_p + \mu_p^T \Sigma_p^{-1} \mu_p)} \frac{1}{\sqrt{(2\pi)^{d_z} |\Sigma_z|}} e^{-\frac{1}{2}\|z - \mu_z\|_{\Sigma_z}^2}. \quad (\text{N.22})$$

Now let us use the fact that $\mathbb{P}(z|H^-)$ is a valid pdf over z

$$\begin{aligned} & \int \mathbb{P}(z|H^-) dz \\ &= \int \sqrt{\frac{|\Sigma_x||\Sigma_z|}{|\Sigma_p||\Sigma_v|}} e^{-\frac{1}{2}(-\mu_z^T \Sigma_z^{-1} \mu_z - \mu_p^T \Sigma_p^{-1} \Sigma_x \Sigma_p^{-1} \mu_p + \mu_p^T \Sigma_p^{-1} \mu_p)} \frac{1}{\sqrt{(2\pi)^{d_z} |\Sigma_z|}} e^{-\frac{1}{2}\|z - \mu_z\|_{\Sigma_z}^2} dz \quad (\text{N.23}) \end{aligned}$$

$$= \sqrt{\frac{|\Sigma_x||\Sigma_z|}{|\Sigma_p||\Sigma_v|}} e^{-\frac{1}{2}(-\mu_z^T \Sigma_z^{-1} \mu_z - \mu_p^T \Sigma_p^{-1} \Sigma_x \Sigma_p^{-1} \mu_p + \mu_p^T \Sigma_p^{-1} \mu_p)} \int \frac{1}{\sqrt{(2\pi)^{d_z} |\Sigma_z|}} e^{-\frac{1}{2}\|z - \mu_z\|_{\Sigma_z}^2} dz \quad (\text{N.24})$$

$$= \sqrt{\frac{|\Sigma_x||\Sigma_z|}{|\Sigma_p||\Sigma_v|}} e^{-\frac{1}{2}(\mu_p^T (\Sigma_p^{-1} - \Sigma_p^{-1} \Sigma_x \Sigma_p^{-1}) \mu_p - \mu_z^T \Sigma_z^{-1} \mu_z)} = 1 \quad (\text{N.25})$$

using the matrix determinant lemma over $|\Sigma_x|$ we get

$$\begin{aligned} \sqrt{\frac{|\Sigma_x||\Sigma_z|}{|\Sigma_p||\Sigma_v|}} &= \sqrt{\frac{|\Sigma_v + H\Sigma_p H^T||\Sigma_p^{-1} + H^T \Sigma_v^{-1} H|^{-1}}{|\Sigma_p||\Sigma_v|}} \\ &= \sqrt{\frac{|\Sigma_p^{-1} + H^T \Sigma_v^{-1} H||\Sigma_p||\Sigma_v|}{|\Sigma_p^{-1} + H^T \Sigma_v^{-1} H||\Sigma_p||\Sigma_v|}} = 1, \quad (\text{N.26}) \end{aligned}$$

meaning the following has to hold

$$\mu_p^T (\Sigma_p^{-1} - \Sigma_p^{-1} \Sigma_x \Sigma_p^{-1}) \mu_p = \mu_z^T \Sigma_z^{-1} \mu_z. \quad (\text{N.27})$$

And so

$$\mathbb{P}(z|H^-) = \mathcal{N}(\mu_z, \Sigma_z) \quad (\text{N.28})$$

where

$$\Sigma_z \doteq \Sigma_v + H\Sigma_p H^T = \Sigma_v + \mathcal{H}\Sigma_w \mathcal{H}^T + \mathcal{H}\mathcal{F}\Sigma_o \mathcal{F}^T \mathcal{H}^T \quad (\text{N.29})$$

$$\mu_z \doteq (\Sigma_v + H\Sigma_p H^T) \Sigma_v^{-1} H \Sigma_x \Sigma_p^{-1} \mu_p = \mathcal{H}\mathcal{F}\mu_o + \mathcal{H}\mathcal{J}\mu_u. \quad (\text{N.30})$$

*Procrastination is the art
of keeping up with yesterday.*

Don Marquis

O

Gaussian Quadratic as χ^2

Lemma 7. Let Δ be quadratic expression $\Delta = \mathcal{S}^T C \mathcal{S} + c^T \mathcal{S} + y$, where $C^T = C$, $\mathcal{S} \sim \mathcal{N}(\mu, \Sigma)$, and $\Sigma > 0$. Then the set of necessary and sufficient conditions for Δ to be distributed as non-central χ^2 with non-centrality parameter δ^2 and degrees of freedom r is that

$$C\Sigma C = C \quad (\text{O.1})$$

$$r = \text{tr}(C\Sigma) \quad (\text{O.2})$$

$$c = C\Sigma c, \quad y = \frac{1}{4}c^T \Sigma c, \quad \delta^2 = \mu^T C\mu + \mu^T c + y \quad (\text{O.3})$$

Proof. This is Theorem 5.1.4 in [52]. □

באופן דומה, קבלת החלטות תחת אי וודאות תוך התבוססות על תכוני עבר (BSP-X) מבצעת את תהליך קבלת החלטות באופן אינקרמנטלי ע"י עדכון מושכל של עצי תכנון הזמינים מטהליכי קבלת החלטות שכבר הושלמו.

אנחנו מציעים גישה שמהווה שינוי עקרוני לגישות הקיימות כיום, עדכון ההסקה הסתברותית בהסתמך על תוצרים ממרחוב האמונה הסתברותי. הגישה שלנו קוראת תיגר על המיציות הקיימות ביום בין תכנון להסקה הסתברותית ומדגימה שבديוק כמו המוח האנושי, שימוש בתהילכים שבוצעו בתהליך קבלת ההחלטה לטובת תהליך ההסקה מאפשרים ל��ר מושעויות זמני חישוב יקרים ללא צורך בהקורת בדיקת הפתרון.

חלק מהמחקר בדקנו את הגישות שלנו במגוון תרחישים, תוך השוואתן לגישות המתקדמות ביותר הקיימות כיום. ההשוואה נעשתה הן על בסיס מידע סימולטיבי והן על בסיס מידע אמיתי, במגוון תרחישים אשר המשותף לכלם הינו סבביה לא ידועה ולא וודאית וקטור מצב מממד גבוה. צפוי, הגישות שלנו מדגימות יתרון חישובי משמעותי ביחס לגישה הנוכחית ביום, ללא כל פגיעה בבדיקה הפתרון.

הטמעת הגישות החדשניות שלנו בתהיליכי קבלת החלטות תחת אי וודאות והסקה הסתברותית תאפשר ליעיל פוטנציאלית את זמני החישוב של כל בינה מלאכותית או מערכת אוטונומית, בין אם היא עשויה שימוש בפתרונות מקורבים ובין אם לאו. שינוי התפיסה, לפיה על תהליכי ההסקה ותהליך קבלת ההחלטה להיות מופדרים, פותח פתח לתהיליכי ייעול נוספים בעולם הבינה המלאכותית שיקראו לתיגר על הפרדיגמות המקובלות ביום מחד ויקדמו את הבינה המלאכותית צעד נוספת קדימה מאידך.

שיעור ותכנון הדרגתית אחד עבור ניוט אוטונומי מקורי

תקציר

מערכות אוטונומיות ובינה מלאכותית כבר מזמן חדלו להיות נושאים הבלעדיים לעולם המדע הבדיוני והפכו למציאות בר קיימא. רבים מגדירים את התקופה הנוכחית כ- "טור הזהב" של בינה מלאכותית לאור הפיתוחים וההתקדמות הרבה בתחום בעשור האחרון.

מערכות אוטונומיות ובינה מלאכותית מערבים (סוקן) שנדרש לבצע ביעילות וברמת אמינות גבוהה מגוון משימות תוך הימצאות בסביבה עם שפע של מקומות אי-ודאות, לחוב לא מידע מוקדים או עם מידע מוגבל על אותה סביבה. תנאים אלו מצריכים את המערכת האוטונומית לבצע הסקה הסתברותית על מגוון רחב של משתנים הקשורים למשימה ולתפקוד המערכת.

כל מערכת אוטונומית או בינה מלאכותית חייבת להכיל שני תהליכי בסיסיים עוקבים ומחזוריים, שעורר וקובלת החלטות תחת אי-ודאות. אם ניקח דוגמא מעולם הננווט, רובוט שנדרש להגיע ליעד מסוים בסביבה לא ידועה צריך לאסוף מידע על הסביבה ולעבד אותו (הסקה הסתברותית), לבצע תהליך של תכנון לגבי המשך דרכו, לקבל החלטה תחת אי-ודאות לגבי הסביבה ומצבו בסביבה, לישם את אותה החלטה,שוב לאסוף מידע מהסביבה וחזור חלילה. בכך להיום, תהליך שכזה, במיוחד בסביבה לא מוכרת תחת אי-ודאות עם מספר רב של משתנים, הינו יקר מאוד חישובי ומצבו בסביבה, למירוץ מושך תפקוד בזמן אמיתי. על מנת לאפשר עובודה בזמן אמיתי, משתמשים כיום בשיטות מוקרותות שמקירות את דיקט הפתרון לטובת שיפור זמני החישוב.

בשנים האחרונות הדמיין בין תהליכי השעורר וקובלת החלטות תחת אי-ודאות יצרו עניין מחקרי רב, החל בניסיונות לפתח מסגרת חישובית אחת והcola בבחינת הדואליות בין שני התהליכים. לмерות המאמצים בתחום, תהליכי השעורר והבקרה, או שעורר וקובלת החלטות תחת אי-ודאות עדום נתפסים כשני תהליכי נפרדים.

חלק מהמחקר שלנו, נתקלנו באירועים מובניאים בגישה הקיימת כוון לביצוע התהליכים הללו. המוטיבציה וההשראה לגישת המחקר שלנו שגוראת תיגר על המחייבות הקיימות כוון בין תכנון להסקה הסתברותית לקויה ממערכת הבינה האוטונומית המוכרת לנו – המוח האנושי. ברוח זו אנו מציעים גישה חדשה המתיחסת לשעורר ותכנון הדרגי כחלקים משותפים באותה המערכת (המכונה Qo).

דרך השיטופיות המתאפשרת תחת השימוש ב- Qo, פיתחנו שתי גישות חדשות לעדכון ההסקה ההסתברותית וקובלת החלטות תחת אי-ודאות. הגישות הללו מהוות שינוי עקרוני לגישות הקיימות כוון, עדכון ההסקה ההסתברותית בהסתמך על תוצרים ממוחב האמונה ההסתברותי (המכונה RUBI) וקובלת החלטות תחת אי-ודאות תוך התבוסת על תכינוי עבר (המכונה BSP-X).

עדכון ההסקה ההסתברותית בהסתמך על תוצרים ממוחב האמונה ההסתברותי (RUBI) מאפשרת לקצר משמעותית זמני חישוב יקרים ללא צורך בהקרבת דיקט הפתרון, באמצעות שימוש בתהליכי וחישובים שכבר בוצעו בתהליך קובלת החלטות המוקדים.

המחקר נעשה בהנחייתו של פרופסור אדים איינדמן במסגרת התוכנית הבין ייחידית למערכות אוטונומיות ורוביוטיקה בטכניון.

אני מודה לטכניון על התמיכה הכספית הנדיבה בהשתלמותי.

שיעור ות鹺ן הדרגתית אחד עבור ניוט אוטונומי מוקוון

חיבור על מחקר

לשם מילוי חלקו של הדרישות לקבלת התואר
דוקטור לפילוסופיה

אלעד פרחי

הוגש לסנט הטכניון – מכון טכנולוגי לישראל
אדר תשפ"א, חיפה פברואר 2021

שיעור ות鹺ן הדרגתית אחוד עבור ניוט אוטונומי מקוון

אלעד פרחי