

# Information-Theoretic Analysis of Non-Cooperative and Cooperative Remote Sensing

Dissertation zur Erlangung des akademischen Grades

Doktor-Ingenieur (Dr.-Ing.)

der Fakultät für Informatik und Elektrotechnik der Universität Rostock

eingereicht von

Steffen Steiner | geb. am 18.07.1987 in Lüneburg

Datum der Einreichung: 24.03.2023

Datum der Verteidigung: 13.09.2023

**Gutachter:**

Volker Kühn

Universität Rostock

Albert-Einstein-Str. 26

18059 Rostock

**Zweitgutachterin:**

Anja Klein

Technische Universität Darmstadt

Landgraf-Georg-Str. 4

64283 Darmstadt



Dieses Werk ist lizenziert unter einer  
Creative Commons Namensnennung 4.0 International Lizenz.

## Danksagung

An erster Stelle möchte ich meinem Doktorvater Prof. Dr.-Ing Volker Kühn für die Betreuung meiner Arbeit danken. Dazu gehört vor allem die fachliche, wissenschaftliche und methodische aber auch moralische Unterstützung, die ich während meiner gesamten Zeit am Institut für Nachrichtentechnik der Universität Rostock genossen habe. Ohne diese Unterstützung wäre die Anfertigung dieser Promotionsschrift niemals zustande gekommen.

Darüber hinaus geht mein Dank an meine Arbeitskollegen, die meine Zeit am Institut für Nachrichtentechnik in einer familiären Atmosphäre unvergesslich gemacht haben. Insbesondere danke ich meinem Bürokollegen und guten Freund Clemens Konrad-Müller für die vielen fachlichen und allgemeinen Diskussionen und Ratschläge sowie den Anmerkungen und Korrekturen für die Erstellung dieser Arbeit und anderen wissenschaftlichen Publikationen. Mein Dank geht ebenso an Dr.-Ing. Henryk Richter für die fachliche Unterstützung in vielen Bereichen auch Abseits des Themas dieser Arbeit.

Ein besonderer Dank geht auch an Dr.-Ing. Maximilian Stark für die Zusammenarbeit sowie die vielen fachlichen Diskussionen und Anregungen, speziell im Bereich von Information Bottleneck.

Zu guter Letzt danke ich meiner Familie, insbesondere meiner Lebensgefährtin Leonie Kandler für die stetige Unterstützung und Motivation sowie für den nötigen Freiraum und die Zeit diese Arbeit anzufertigen.

## Eidesstattliche Erklärung

Hiermit versichere ich an Eides statt, dass ich diese Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe. Alle Stellen, die wörtlich oder auch sinngemäß aus Veröffentlichungen entnommen sind, habe ich als solche kenntlich gemacht. Ich weiß, dass bei Abgabe einer falschen Versicherung die Prüfung als nicht bestanden zu gelten hat.

**Steffen Steiner**, Rostock, 24.03.2023 und 18.10.2023

# Zusammenfassung

Diese Arbeit untersucht die Optimierung verteilter Kompression in Sensornetzwerken. Dabei wird der Spezialfall betrachtet, bei dem einzelne Sensoren verrauschte Beobachtungen von dem selben Prozess machen und diese über kapazitätsbegrenzte Kanäle an einen gemeinsamen Empfänger weiterleiten wollen. Sollte die Kommunikation unter den Sensoren nicht möglich sein, ist dies unter dem Namen Chief Executive (Estimation) Officer (CEO) Problem bekannt. Um die Beobachtungen mit möglichst großer Rate fehlerfrei übertragen zu können, muss jeder Sensor eine lokale Komprimierung vornehmen. Dieses Problem stellt ein altbekanntes Problem in der Informationstheorie dar, dessen Untersuchung in den letzten Jahren signifikante Ergebnisse erzielt hat. Insbesondere wurde die komplette Region erreichbarer Übertragungsraten für verschiedene Verteilungen involvierter Zufallsprozesse sowie Verzerrungsmaße bestimmt. Darüber hinaus gibt es verschiedene algorithmische Ansätze, das CEO Problem zu lösen. Die praktische Ausführung dieser Algorithmen kann allerdings sehr anspruchsvoll bzw. komplex werden. Daher befasst sich diese Arbeit mit der praktischen Implementierung und Verfeinerung sowie der informationstheoretischen Analyse solcher Algorithmen. Betrachtet man das CEO Problem mit einem logarithmischen Verzerrungsmaß besteht ein direkter Zusammenhang zu dem sogenannten information bottleneck (IB) Prinzip. Dementsprechend wird zur Lösung des CEO Problems ein gieriger Algorithmus basierend auf diesem Prinzip eingeführt, welcher gleichzeitig individuelle Ratenbeschränkungen der Weiterleitungskanäle zum gemeinsamen Empfänger erlaubt. Hierbei werden die Komprimierer/Quantisierer der einzelnen Sensoren nacheinander optimiert, wobei die Abbildungsvorschriften der Quantisierer vorangegangener Sensoren, entsprechend des Wyner-Ziv Kodierprinzips, ausgenutzt werden. Dieser algorithmische Ansatz liefert eine signifikante Verbesserung der Leistungsfähigkeit verglichen mit einer skalaren und unabhängigen IB Optimierung der einzelnen Sensoren. Allerdings besteht auch bei diesem algorithmischen Ansatz ein exponentieller Zusammenhang zwischen dem Speicherbedarf zur Optimierung und der Größe des Sensornetzwerks. Um dieses Problem zu lösen, wird ein Ansatz verfolgt, bei dem die Abbildungsvorschriften der Quantisierer vorangegangener Sensoren basierend auf dem IB Prinzip weiter komprimiert werden. Dies reduziert den Speicherbedarf signifikant, wohingegen der Kompressionsverlust bezüglich des unkomprimierten Verfahrens gering gehalten wird. Im Falle einer fehlerbehafteten Übertragung zum gemeinsamen Empfänger, was zwangsläufig bei Verwendung von Codes mit endlicher Länge auftritt, wird der gierige Optimierungsalgorithmus adaptiert, um eben diese Fehler zu berücksichtigen.

Verglichen mit dem CEO Szenario mit voller Kooperation, bei dem jedem Sensor die Messwerte aller Sensoren zur Verfügung stehen und eine koordinierte Kompression durchgeführt werden kann, verbleibt bei dem bisherigen Algorithmus für das originale, unkooperierende CEO Problem eine große Lücke bezüglich der Leistungsfähigkeit. Aus diesem Grund wird das partiell kooperierende CEO Problem eingeführt, welches das originale Problem dahingehend erweitert, dass Sensoren über kapazitätsbegrenzte Kanäle

---

zur Laufzeit miteinander kommunizieren können. Dementsprechend kann diese instantane Seiteninformation, basierend auf dem verwendeten Übertragungsprotokoll, bei der Kompression ausgenutzt werden. Die Optimierung der Kompression einzelner Sensoren erfolgt wie beim originalen CEO Problem über einen gierigen Algorithmus. Beim sukzessiven Broadcast-Übertragungsprotokoll kann jeder Sensor die instantane Seiteninformation aller vorher übertragenden Sensoren ausnutzen. Da die Quantisierungsfunktion von der Größe des Netzwerks abhängt, könnte es bei der Nutzung dieses Übertragungsprotokolls zu Speicherproblemen kommen. Um dies zu umgehen, beschränkt sich das sequentielle Punkt-zu-Punkt-Übertragungsprotokoll auf die Kommunikation mit direkten Nachbarn. Dementsprechend kann lediglich die instantane Seiteninformation des direkten Vorgängers genutzt werden. Mit Hilfe des Austauschs von instantaner Seiteninformation zwischen Sensoren kann die Leistungsfähigkeit gegenüber dem originalen CEO Problem signifikant verbessert werden. Darüber hinaus verbessert der Austausch instantaner Seiteninformation die Robustheit gegenüber schlechten Wyner-Ziv Kodierungen, welche durch suboptimale Optimierungsreihenfolgen hervorgerufen werden und deutliche Einbrüche der Leistungsfähigkeit beim originalen CEO Problem mit sich bringen können. Da trotz partieller Kooperation mit sukzessiver Broadcast oder sequentieller Punkt-zu-Punkt Übertragung im Vergleich zum vollkooperierenden CEO Problem, bezüglich der Leistungsfähigkeit, weiterhin eine Lücke bleibt, wird das Zweiphasen-Übertragungsprotokoll eingeführt. Dieses separiert die Kooperation zwischen Sensoren und das Weiterleiten zum gemeinsamen Empfänger in einzelne Phasen und ermöglicht so, dass Sensoren die maximale instantane Seiteninformation ausnutzen können. Mit Hilfe dieses Übertragungsprotokolls kann die Leistungsfähigkeit des CEO Szenarios mit voller Kooperation erreicht werden.

## Abstract

This work addresses the optimization of distributed compression in a sensor network. In particular, distributed sensors measure noisy versions of the same process of interest and try to forward their measurements over capacity-limited links to a common receiver. Therefore, each sensor has to locally compress its measurements. If direct communication among sensors is not possible, this setup is widely known as the Chief Executive (Estimation) Officer (CEO) problem. This problem represents a long-standing problem in information theory, and significant progress has been achieved in recent years. In particular, the region of achievable transmission rates has been completely characterized for specific probability distributions of involved random processes and distortion measures. Moreover, algorithmic solutions to solve the CEO problem are principally known. Their practical implementation, however, can become challenging due to complexity reasons. Therefore, this work focuses on an implementation point of view to solve this problem as well as an information-theoretic analysis of the proposed algorithms. Since the CEO problem with a logarithmic loss distortion measure is closely related to the information bottleneck (IB) principle, an efficient greedy algorithm based on this principle is introduced to determine feasible solutions of the CEO problem while fulfilling individual rate constraints of the forward link of each sensor. Applying Wyner-Ziv coding, the compression devices/quantizers are successively designed, exploiting the quantizer mappings of already designed sensors. This algorithmic approach leads to a significant performance gain compared to individual scalar IB optimization of each sensor. However, even this greedy optimization algorithm results in an exponentially growing memory complexity with the number of sensors in the network, which is why an IB based approach to compress the exploited quantizer mappings for Wyner-Ziv coding is introduced. This significantly reduces the memory requirements, while the loss compared to the uncompressed case can be kept very small. For the case of erroneous transmission to the common receiver caused, e.g., by finite length codes, the greedy optimization algorithm is adapted, incorporating residual error probabilities on the forward channel of each sensor.

The greedy optimization approach still exhibits a significant gap to a fully cooperative CEO scenario, where each sensor has access to all measurements in the network, and coordinated compression can be performed. Therefore, this work introduces the partial cooperative CEO scenario, extending the original CEO problem by allowing sensors to communicate over capacity-limited inter-sensor links during run-time. Hence, sensors can exploit instantaneous side-information received from other sensors for compressing their measurements depending on the specific inter-sensor communication protocol. Inspired by algorithmic solutions for the original CEO problem, the sensors are optimized in a greedy manner. The successive broadcast transmission protocol allows each sensor to exploit instantaneous side-information of all previously transmitting sensors. However, this introduces memory issues for larger networks since the mapping is dependent on the network size. In order to solve this, the sequential point-to-point transmission protocol

---

restricts the inter-sensor communication such that sensors can only exploit instantaneous side-information of the direct predecessor. This inter-sensor communication significantly increases the performance of the CEO scenario. Moreover, exchanging instantaneous side-information increases the robustness against bad Wyner-Ziv coding strategies caused by a specific optimization order, which can lead to significant performance losses in the original CEO problem. Since there still occurs a gap to the fully cooperative CEO scenario, a two-phase transmission protocol is introduced, separating the cooperation among sensors and the forwarding to the common receiver. This transmission protocol can reach the performance of a fully cooperative CEO scenario.

# Contents

<b>List of Figures</b>	<b>X</b>
<b>List of Algorithms</b>	<b>XV</b>
<b>List of Acronyms</b>	<b>XVI</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Structure of Thesis . . . . .	2
1.3 Notations . . . . .	3
<b>2 Fundamentals</b>	<b>4</b>
2.1 Densities and Probabilities . . . . .	4
2.1.1 Continuous Probability Densities . . . . .	4
2.1.2 Discrete Probability Distributions . . . . .	4
2.1.3 Joint Probability Distribution . . . . .	5
2.1.4 Conditional Probability Distribution . . . . .	5
2.1.5 Expectation and Variance . . . . .	6
2.2 Information Theory . . . . .	6
2.2.1 Information and Entropy . . . . .	6
2.2.2 Mutual Information . . . . .	7
2.2.3 Kullback-Leibler divergence . . . . .	9
2.2.4 Jensen-Shannon divergence . . . . .	9
2.3 Channel Models . . . . .	10
2.3.1 Channel Capacity . . . . .	10
2.3.2 Additive White Gaussian Noise Channel . . . . .	10
2.3.3 Discrete Memoryless Channel . . . . .	12
2.4 Discussion . . . . .	12
<b>3 Preliminaries on Lossy Compression</b>	<b>14</b>
3.1 Rate-Distortion Theory . . . . .	14
3.1.1 Rate-Distortion Function and Distortion-Rate Function . . . . .	16
3.1.2 Blahut-Arimoto Algorithm . . . . .	18
3.2 Wyner-Ziv Coding . . . . .	19
3.3 Remote Sensing (Noisy Source Coding) . . . . .	20
3.4 The Information Bottleneck Method . . . . .	21
3.4.1 The Optimization Problem . . . . .	23
3.4.2 Iterative Information Bottleneck Algorithm . . . . .	24
3.5 Channel-Aware Information Bottleneck Approach . . . . .	32
3.6 Distributed Remote Sensing . . . . .	39



3.6.1	Non-Cooperative Remote Sensing - The CEO Problem . . . . .	39
3.6.2	Fully Cooperative Remote Sensing . . . . .	44
3.7	Discussion . . . . .	46
<b>4</b>	<b>Algorithmic Solutions for Non-Cooperative Distributed Remote Sensing</b>	<b>47</b>
4.1	A Greedy Distributed Information Bottleneck Approach . . . . .	47
4.2	An Algorithmic Solution . . . . .	51
4.2.1	Simulation Setup . . . . .	53
4.2.2	Achievable Rate Tuples . . . . .	53
4.2.3	Contribution of Wyner-Ziv coding . . . . .	54
4.2.4	Influence of Sum-Rate . . . . .	55
4.2.5	Influence of the Network Size . . . . .	56
4.2.6	Influence of the Relevant Signal . . . . .	59
4.2.7	Influence of Optimization Order for Asymmetric Scenarios . . . . .	62
4.2.8	Robustness of the GDIB Algorithm . . . . .	63
4.2.9	Greedy Distributed Information Bottleneck Algorithm for Deterministic Mappings . . . . .	68
4.3	Reduced-Memory Complexity GDIB Algorithm . . . . .	72
4.3.1	Sequential Compression Scheme . . . . .	73
4.3.2	One-Step Compression Scheme . . . . .	74
4.3.3	Analysis of Compression Schemes . . . . .	75
4.3.4	Performance of Reduced-Memory Complexity GDIB . . . . .	76
4.4	Distributed Channel Aware Information Bottleneck Algorithm . . . . .	80
4.5	Discussion . . . . .	88
<b>5</b>	<b>Distributed Remote Sensing with Partial Cooperation</b>	<b>90</b>
5.1	Partially Cooperative CEO Scenario . . . . .	90
5.2	Successive Broadcast Transmission Protocol . . . . .	92
5.2.1	Generation of Broadcast Side-Information . . . . .	94
5.2.2	Generate Information Forwarded to the Common Receiver . . . . .	97
5.3	Sequential Point-To-Point Protocol . . . . .	101
5.3.1	Generation of Point-To-Point Side-Information . . . . .	102
5.3.2	Generate Information Forwarded to the Common Receiver . . . . .	106
5.4	Two-Phase Transmission Protocol with Artificial Side-Information . . . . .	114
5.5	Discussion . . . . .	117
<b>6</b>	<b>Conclusion</b>	<b>119</b>
	<b>Bibliography</b>	<b>122</b>
	<b>Own Publications</b>	<b>134</b>
	<b>Appendices</b>	<b>135</b>
<b>A</b>	<b>Proof for Supermodularity of <math>I(\mathcal{Y}_S; \mathcal{Z}_S   \mathcal{Z}_S, \mathcal{Q})</math></b>	<b>136</b>

<b>B Derivation of the GDIB Algorithm</b>	<b>137</b>
B.1 Derivative of $I(\mathcal{X}; \mathcal{Z}_m   \mathcal{Z}_{<m})$ . . . . .	137
B.2 Derivative of $I(\mathcal{Y}_m; \mathcal{Z}_m   \mathcal{Z}_{<m})$ . . . . .	138
B.3 Fusion of Derived Parts . . . . .	139
B.4 Calculating required pmfs . . . . .	139
 <b>C Derivation of an IB-Based Algorithm to Generate Broadcasting Side-Information</b>	 <b>141</b>
C.1 Derivative of $I(\mathcal{X}; \mathcal{S}_m   \mathcal{S}_{<m})$ . . . . .	141
C.2 Derivative of $I(\mathcal{Y}_m; \mathcal{S}_m   \mathcal{S}_{<m})$ . . . . .	142
C.3 Fusion of Derived Parts . . . . .	143
C.4 Calculating required pmfs . . . . .	143
 <b>D Derivation of a GDIB Algorithm for Cooperating Sensors Applying the Successive Broadcasting Protocol</b>	 <b>145</b>
D.1 Derivative of $I(\mathcal{X}; \mathcal{Z}_m   \mathcal{Z}_{<m})$ . . . . .	145
D.2 Derivative of $I(\mathcal{Y}_m, \mathcal{S}_{<m}; \mathcal{Z}_m   \mathcal{Z}_{<m})$ . . . . .	146
D.3 Fusion of Derived Parts . . . . .	147
D.4 Calculating required pmfs . . . . .	148
 <b>E Derivation of an IB-Based Algorithm to Generate Point-To-Point Side-Information</b>	 <b>149</b>
E.1 Derivative of $I(\mathcal{X}; \mathcal{S}_m)$ . . . . .	149
E.2 Derivative of $I(\mathcal{Y}_m, \mathcal{S}_{m-1}; \mathcal{S}_m)$ . . . . .	150
E.3 Fusion of Derived Parts . . . . .	151
E.4 Calculating required pmfs . . . . .	151
 <b>F Derivation of a GDIB Algorithm for Cooperating Sensors Applying the Sequential Point-To-Point Protocol</b>	 <b>153</b>
F.1 Derivative of $I(\mathcal{X}; \mathcal{Z}_m   \mathcal{Z}_{<m})$ . . . . .	153
F.2 Derivative of $I(\mathcal{Y}_m, \mathcal{S}_{m-1}; \mathcal{Z}_m   \mathcal{Z}_{<m})$ . . . . .	154
F.3 Fusion of Derived Parts . . . . .	155
F.4 Calculating required probability mass functions (pmfs) . . . . .	156

## List of Figures

2.1	Schematic representation of an additive white Gaussian noise (AWGN) channel . . . . .	10
2.2	Capacity and spectral efficiency of an AWGN channel for different digital modulation schemes . . . . .	11
2.3	Schematic representation of an discrete memoryless channel (DMC) channel	12
3.1	Schematic illustration of lossy compression [EK11] . . . . .	14
3.2	Encoding process for lossy source coding . . . . .	15
3.3	Rate-distortion function $R(D)$ for a Gaussian source $y \sim \mathcal{N}(0, 1)$ and squared Euclidean distance distortion measure $\tilde{d}(y, z) =  \hat{y}(z) - y ^2$ showing achievable and non-achievable regions [CT06] . . . . .	18
3.4	Illustration of the convex optimization process applying the Blahut-Arimoto algorithm in $\mathbb{R}^2$ [CT06; Slo02] . . . . .	18
3.5	Wyner-Ziv system model - source coding with side-information . . . . .	19
3.6	General remote sensing setup - encoder compresses noisy signal to forward it over a channel with capacity $C$ to a receiver . . . . .	20
3.7	Illustration of the IB optimization problem over three convex sets applying the iterative IB algorithm [Slo02] . . . . .	25
3.8	Relevance-Compression plane using the iterative IB algorithm for different cardinalities $ \mathbb{Z} $ , $ \mathbb{X}  = 4$ , $ \mathbb{Y}  = 256$ , SNR $\gamma = 6$ dB . . . . .	28
3.9	Relevance-Compression plane comparing stochastic iterative IB and the deterministic iterative IB algorithm for different $ \mathbb{X} $ , $ \mathbb{Y}  = 512$ and SNR $\gamma = \{6, 15\}$ dB . . . . .	29
3.10	Relevant mutual information in the case of measurement signal-to-noise ratio (SNR) mismatch using the IB method with $ \mathbb{X}  = 4$ , $ \mathbb{Y}  = 512$ and $ \mathbb{Z}  = 4$ . . . . .	30
3.11	Compression rate in the case of measurement SNR mismatch using the IB method with $ \mathbb{X}  = 4$ , $ \mathbb{Y}  = 512$ and $ \mathbb{Z}  = 4$ . . . . .	31
3.12	Schematic illustration of the Channel Aware Information Bottleneck (CAIB) setup modelling residual error probabilities as a DMC . . . . .	33
3.13	Relevant mutual information $I(\mathcal{X}; \tilde{\mathcal{Z}})$ for different measurement signal-to-noise ratios and an artificially modeled DMC comparing the Channel-Aware Information Bottleneck to the original IB approach; $ \mathbb{X}  = 4$ , $ \mathbb{Y}  = 64$ , $ \mathbb{Z}  = 8$	35
3.14	Relevant mutual information $I(\mathcal{X}; \tilde{\mathcal{Z}})$ for different measurement signal-to-noise ratios and an AWGN modeled DMC comparing the Channel-Aware Information Bottleneck to the original IB approach; $ \mathbb{X}  = 4$ , $ \mathbb{Y}  = 64$ , $ \mathbb{Z}  = 8$	36
3.15	Measurement channel mismatch applying the CAIB algorithm with $ \mathbb{X}  = 4$ , $ \mathbb{Y}  = 512$ , $ \mathbb{Z}  = 8$ . . . . .	37

---

3.16	Forward channel mismatch applying the CAIB algorithm with $ \mathbb{X}  = 4$ , $ \mathbb{Y}  = 512$ , $ \mathbb{Z}  = 8$ . . . . .	38
3.17	System model for the non-cooperative remote sensing problem . . . . .	39
3.18	Encoding process the non-cooperative remote sensing problem . . . . .	40
3.19	Illustration of the Markov property of the CEO system model . . . . .	40
3.20	System model for the fully cooperative remote sensing scenario modeled by a centralized compression approach . . . . .	44
3.21	Encoder for the fully cooperative remote sensing problem . . . . .	45
4.1	Flowchart of the overall Greedy Distributed Information Bottleneck (GDIB) algorithm . . . . .	53
4.2	Achievable rate tuples for different relevant mutual information using the GDIB algorithm in a symmetric scenario with two sensors, $ \mathbb{X}  = 4$ , $ \mathbb{Z}_m  = 4$ and SNRs $\gamma_m = 8$ dB . . . . .	54
4.3	Contribution of the second sensor applying the GDIB algorithm in a sym- metric scenario with two sensors, $ \mathbb{X}  = 4$ , $ \mathbb{Z}_m  = 4$ and SNRs $\gamma_m = 8$ dB . . . . .	55
4.4	Relevant mutual information vs. sum capacity for a symmetric scenario with $M = 5$ sensors, $ \mathbb{X}  = 4$ , SNRs $\gamma_m \in \{3, 8\}$ dB and $ \mathbb{Z}_m  = 4$ . . . . .	56
4.5	Relevant mutual information vs. number of sensors for fixed a sum-rate $C_{\text{sum}} = 2.5$ bit/s/Hz, $ \mathbb{X}  = 4$ , $ \mathbb{Z}_m  = 4$ . . . . .	57
4.6	Relevant mutual information vs. number of sensors for fixed a sum-rate $C_{\text{sum}} = 4.0$ bit/s/Hz, $ \mathbb{X}  = 4$ , $ \mathbb{Z}_m  = 4$ . . . . .	57
4.7	Relevant mutual information vs. number of sensors for fixed individual link capacity $C_m = 0.5$ bit/s/Hz and $ \mathbb{X}  = 4$ , $ \mathbb{Z}_m  = 4$ . . . . .	58
4.8	Relevant mutual information vs. number of sensors for fixed individual link capacity $C_m = 1$ bit/s/Hz and $ \mathbb{X}  = 4$ , $ \mathbb{Z}_m  = 4$ . . . . .	59
4.9	Relevant mutual information vs. sum capacity for a symmetric scenario with $M = 5$ sensors, SNRs $\gamma_m = 3$ dB, $ \mathbb{Z}_m  = 4$ and a Gaussian relevant signal $x$ with $ \mathbb{X}  = 64$ . . . . .	60
4.10	Relevant mutual information vs. number of sensors for a fixed sum-rate $C_{\text{sum}} \in \{2.5, 4.0\}$ bit/s/Hz, $ \mathbb{Z}_m  = 4$ , $\gamma_m = 3$ dB and a Gaussian relevant signal $x$ with $ \mathbb{X}  = 64$ . . . . .	61
4.11	Relevant mutual information vs. number of sensors after GDIB optimization for a fixed sum-rate $C_{\text{sum}} = 4.0$ bit/s/Hz, $ \mathbb{Z}_m  = 16$ . . . . .	61
4.12	Relevant mutual information for an asymmetric scenario with $M = 4$ sen- sors, SNRs $\gamma_m = [2, 4, 6, 8]$ dB and $ \mathbb{X}  = 4$ using GDIB optimization . . . . .	63
4.13	Measurement channel mismatch applying the GDIB algorithm for $M = 5$ sensors with $C_{\text{sum}} = 2.5$ bit/s/Hz equally distributed on each forward link $C_m = \frac{C_{\text{sum}}}{M}$ , $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 512$ and $ \mathbb{Z}_m  = 4$ . . . . .	65
4.14	Measurement channel mismatch applying the GDIB algorithm for $M = 5$ sensors with $C_{\text{sum}} = 5.0$ bit/s/Hz equally distributed on each forward link $C_m = \frac{C_{\text{sum}}}{M}$ , $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 512$ and $ \mathbb{Z}_m  = 4$ . . . . .	66

---

4.15	Measurement channel mismatch applying the GDIB algorithm for $M = 5$ sensors with $C_{\text{sum}} = 7.5$ bit/s/Hz equally distributed on each forward link $C_m = \frac{C_{\text{sum}}}{M}$ , $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 512$ and $ \mathbb{Z}_m  = 4$ . . . . .	67
4.16	Measurement channel mismatch applying the GDIB algorithm for $M = 5$ sensors with $C_{\text{sum}} = 10.0$ bit/s/Hz equally distributed on each forward link $C_m = \frac{C_{\text{sum}}}{M}$ , $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 512$ and $ \mathbb{Z}_m  = 4$ . . . . .	67
4.17	Flowchart of the deterministic GDIB algorithm . . . . .	69
4.18	Compression-Relevance curve for $M = 3$ , $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 64$ and $\gamma_m = 8$ dB; for stochastic GDIB: $ \mathbb{Z}_m  = 16$ . . . . .	69
4.19	Comparison of original GDIB and deterministic GDIB with individual link capacities $C_m = \frac{C_{\text{sum}}}{M}$ for different sum-rates; $M = 5$ , $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 64$ and $\gamma_m = 6$ dB; for stochastic GDIB: $ \mathbb{Z}_m  = 16$ . . . . .	71
4.20	(a) sequential compression of $\mathbf{z}_{<6}$ for sensor 6 (b) one-step compression . . .	73
4.21	Evolution of the compressed version $\mathcal{Z}_{<n}^*$ versus sequential compression steps considering the last sensor; symmetric scenario: $M = 10$ sensors, sum-rate $C_{\text{sum}} = 2.5$ bit/s/Hz and $C_m = \frac{C_{\text{sum}}}{M}$ , SNRs $\gamma_m = 8$ dB, $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 64$ and $ \mathbb{Z}_m  = 4$ ; Not Compressed (NC), One-Step Compression (OC), Sequential Compression (SC) . . . . .	76
4.22	Relevant mutual information versus number of sensors; symmetric scenario, sum-rate $C_{\text{sum}} = 2.5$ bit/s/Hz and $C_m = \frac{C_{\text{sum}}}{M}$ , SNRs $\gamma_m = 8$ dB, $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 64$ and $ \mathbb{Z}_m  = 4$ ; One-Step Compression (OC), Sequential Compression (SC) . . . . .	77
4.23	Memory requirements to store $p(x y_{12}, z_{<12}^*)$ in a network of $M = 12$ sensors when optimizing the last one . . . . .	78
4.24	Relevant mutual information versus SNR $\gamma_m$ in dB for symmetric scenario and sum-rate $C_{\text{sum}} = 2.5$ bit/s/Hz, $M = 7$ , $ \mathbb{Y}_m  = 64$ and $ \mathbb{Z}_m  = 4$ ; One-Step Compression (OC), Sequential Compression (SC) . . . . .	78
4.25	Compression loss $\Delta I(\mathcal{X}; \mathcal{Z})$ versus $ \mathbb{Z}^* $ ; Symmetric scenario with $M = 12$ , sum-rate $C_{\text{sum}} = 2.5$ bit/s/Hz, SNRs $\gamma_m = 8$ dB, $ \mathbb{X}  = \{2, 4, 8\}$ , $ \mathbb{Y}_m  = 64$ and $ \mathbb{Z}_m  = 4$ ; One-Step Compression (OC), Sequential Compression (SC) .	79
4.26	Relevant mutual information $I(\mathcal{X}; \tilde{\mathcal{Z}})$ for different measurement signal-to-noise ratios and an artificially modeled DMC comparing the Channel Aware Greedy Distributed Information Bottleneck (CA-GDIB) algorithm to the GDIB approach for a network size of $M = 5$ ; $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 32$ , $ \mathbb{Z}_m  = 8$ .	84
4.27	Relevant mutual information $I(\mathcal{X}; \tilde{\mathcal{Z}})$ for network sizes and an artificially modeled DMC comparing the CA-GDIB algorithm to the GDIB approach; $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 32$ , $ \mathbb{Z}_m  = 8$ . . . . .	85
4.28	Measurement channel mismatch applying the CA-GDIB algorithm with $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 512$ , $ \mathbb{Z}_m  = 8$ and a network with $M = 3$ sensors . . . . .	86
4.29	Forward channel mismatch applying the CA-GDIB algorithm with $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 512$ , $ \mathbb{Z}_m  = 8$ and a network with $M = 3$ sensors . . . . .	87

---

5.1	System model for the partially cooperative Chief Executive Officer (pcCEO) scenario using the successive broadcast transmission protocol . . . . .	93
5.2	Markov model of statistical dependencies for the successive broadcast transmission protocol . . . . .	93
5.3	Encoding process for successive broadcast transmission protocol . . . . .	94
5.4	Graphical illustration of IB fusion of two inputs to determine instantaneous side-information $s_m$ (a) and the quantizer sensor output $z_m$ (b) for the successive broadcast transmission protocol . . . . .	94
5.5	Evolution of side-information $I(\mathcal{X}; \mathcal{S}_{\leq m})$ for sensor $m$ in a network with $M = 6$ sensors and different cardinalities $ \mathcal{S}_m $ for the successive broadcasting protocol; $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 64$ . . . . .	97
5.6	Relevant mutual information $I(\mathcal{X}; \mathcal{Z})$ versus the network size for a fixed sum-rate of $C_{\text{sum}} = 2.5$ bit/s/Hz and $C_m = \frac{C_{\text{sum}}}{M}$ using the successive broadcasting protocol with different cardinalities $ \mathcal{S}_m $ ; $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 64$ , $ \mathbb{Z}_m  = 4$ . . . . .	100
5.7	Relevant mutual information $I(\mathcal{X}; \mathcal{Z})$ versus the network size for a fixed sum-rate of $C_{\text{sum}} = 4$ bit/s/Hz and $C_m = \frac{C_{\text{sum}}}{M}$ using the successive broadcasting protocol with different cardinalities $ \mathcal{S}_m $ ; $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 64$ , $ \mathbb{Z}_m  = 4$ . . . . .	100
5.8	System model for the pcCEO scenario using the sequential point-to-point transmission protocol . . . . .	102
5.9	Markov model of statistical dependencies for the sequential point-to-point transmission protocol . . . . .	103
5.10	Encoding process for the sequential point-to-point transmission protocol . . . . .	103
5.11	Graphical illustration of IB fusion of two inputs to determine instantaneous side-information $s_m$ (a) and the quantizer sensor output $z_m$ (b) for the sequential point-to-point transmission protocol . . . . .	104
5.12	Evolution of side-information $I(\mathcal{X}; \mathcal{S}_m)$ for sensor $m$ in a network with $M = 6$ sensors for different cardinalities $ \mathcal{S}_m $ for the sequential point-to-point transmission protocol; $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 64$ . . . . .	106
5.13	Relevant mutual information $I(\mathcal{X}; \mathcal{Z})$ versus the network size for a fixed sum-rate of $C_{\text{sum}} = 2.5$ bit/s/Hz and $C_m = \frac{C_{\text{sum}}}{M}$ for the sequential point-to-point transmission protocol with different cardinalities $ \mathcal{S}_m $ ; $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 64$ , $ \mathbb{Z}_m  = 4$ . . . . .	108
5.14	Relevant mutual information $I(\mathcal{X}; \mathcal{Z})$ versus the network size for a fixed sum-rate of $C_{\text{sum}} = 4$ bit/s/Hz and $C_m = \frac{C_{\text{sum}}}{M}$ for the sequential point-to-point transmission protocol with different cardinalities $ \mathcal{S}_m $ ; $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 64$ , $ \mathbb{Z}_m  = 4$ . . . . .	110
5.15	Relevant mutual information $I(\mathcal{X}; \mathcal{Z})$ versus sum-rate $C_{\text{sum}}$ with $C_m = \frac{C_{\text{sum}}}{M}$ using the sequential point-to-point transmission protocol with different cardinalities $ \mathcal{S}_m $ ; $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 64$ , $ \mathbb{Z}_m  = 4$ . . . . .	111
5.16	Relevant mutual information for an asymmetric scenario with $M = 4$ sensors, SNRs $\gamma_m = [2, 4, 6, 8]$ dB and $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 64$ , $ \mathbb{Z}_m  = 4$ using the sequential point-to-point transmission protocol with $ \mathcal{S}_m  = 8$ . . . . .	112

---

---

5.17	Relevant mutual information $I(\mathcal{X}; \mathcal{Z})$ versus the network size for a fixed sum-rate of $C_{\text{sum}} \in \{2.5, 4\}$ bit/s/Hz, $\gamma_m = 3$ dB and $C_m = \frac{C_{\text{sum}}}{M}$ for the sequential point-to-point transmission protocol with different cardinalities $ \mathcal{S}_m $ and a Gaussian relevant signal $x$ ; $ \mathbb{X}  = 64$ , $ \mathbb{Y}_m  = 64$ , $ \mathbb{Z}_m  = 4$ . . . . .	113
5.18	Relevant mutual information $I(\mathcal{X}; \mathcal{Z})$ versus sum-rate $C_{\text{sum}}$ in a network of $M = 5$ sensors with $C_m = \frac{C_{\text{sum}}}{M}$ using the sequential point-to-point transmission protocol with different cardinalities $ \mathcal{S}_m $ and a Gaussian relevant signal with $ \mathbb{X}  = 64$ ; $ \mathbb{Y}_m  = 128$ , $ \mathbb{Z}_m  = 4$ . . . . .	114
5.19	Relevant mutual information $I(\mathcal{X}; \mathcal{Z})$ versus the network size for a fixed sum-rate of $C_{\text{sum}} = 2.5$ bit/s/Hz and $C_m = \frac{C_{\text{sum}}}{M}$ using a two-phase transmission protocol for artificially decoupled extrinsic information with different $\gamma_{\text{ext}}$ ; $\gamma_m = 8$ dB, $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 64$ , $ \mathbb{Z}_m  = 4$ , $ \mathbb{S}^*  = 512$ . . . . .	116
5.20	Relevant mutual information $I(\mathcal{X}; \mathcal{Z})$ versus extrinsic mutual information $I(\mathcal{X}; \mathcal{S}^*)$ for different network sizes and a fixed sum-rate of $C_{\text{sum}} = 2.5$ bit/s/Hz and $C_m = \frac{C_{\text{sum}}}{M}$ and $ \mathbb{X}  = 4$ , $ \mathbb{Y}_m  = 64$ , $ \mathbb{Z}_m  = 4$ . . . . .	117

## List of Algorithms

1	Stochastic Iterative IB algorithm [Slo02] . . . . .	26
2	Deterministic Iterative IB algorithm . . . . .	27
3	Channel-Aware IB algorithm [Win14] . . . . .	34
4	Extended Blahut-Arimoto algorithm . . . . .	52
5	Extended Blahut-Arimoto algorithm for the deterministic GDIB algorithm .	70
6	Extended Blahut-Arimoto Algorithm with Reduced-Memory Complexity . .	74
7	Sequential Compression of $\mathbf{z}_{<3}$ for sensor $m = 3$ . . . . .	75
8	Sequential Compression of $\mathbf{z}_{<m}$ for sensor $m > 3$ . . . . .	75
9	One-Step Compression . . . . .	76
10	Distributed Channel Aware GDIB algorithm . . . . .	83
11	Extended Blahut-Arimoto algorithm to obtain instantaneous side-information for the successive broadcasting protocol . . . . .	96
12	Extended Blahut-Arimoto algorithm for the successive broadcasting protocol	99
13	Extended Blahut-Arimoto algorithm to obtain side-information for the se- quential point-to-point transmission protocol . . . . .	105
14	Extended Blahut-Arimoto algorithm for the sequential point-to-point trans- mission protocol . . . . .	109



# List of Acronyms

**AWGN** additive white Gaussian noise.

**BERM** binary encoding with rate-matching.

**BS** base station.

**CIB** Centralized Information Bottleneck.

**C-RAN** Cloud Radio Access Network.

**CA-GDIB** Channel Aware Greedy Distributed Information Bottleneck.

**CAIB** Channel Aware Information Bottleneck.

**CEO** Chief Executive (Estimation) Officer.

**DMC** discrete memoryless channel.

**EXIT** extrinsic information transfer.

**fcCEO** fully cooperative Chief Executive Officer.

**FFT** fast Fourier transform.

**GDIB** Greedy Distributed Information Bottleneck.

**GDIB-PTP** Greedy Distributed Information Bottleneck-Point-to-Point Side-Information.

**GDIB-BC** Greedy Distributed Information Bottleneck-Broadcast Side-Information.

**IB** information bottleneck.

**JS** Jensen-Shannon.

**KL** Kullback-Leibler.

**LDPC** low density parity check.

**MAC** multiple access channel.

**MC** memoryless channel.

**MSE** mean squared error.

**OFDM** orthogonal frequency division multiplexing.

**pcCEO** partially cooperative Chief Executive Officer.

**pdf** probability density function.

**pmf** probability mass function.

**QoS** quality of service.

**RAP** radio access point.

**RS** remote sensing.

**RU** radio unit.

**SNR** signal-to-noise ratio.

**UE** user equipment.

**WZ** Wyner-Ziv.

# Chapter 1

## Introduction

### 1.1 Motivation

Distributed sensing plays an important role in many areas, such as smart environments, cities, or homes, which shall improve safety and comfort. Environmental monitoring systems, smart manufacturing (Industry 4.0), or systems for driving assistance in cars employ many sensors and fuse their measurements to infer a signal of interest. In all these scenarios, spatially separated devices collect data and forward it to a central processing unit via capacity-limited communications links. Therefore, sensing, communication, and signal processing must be jointly optimized to infer the desired information efficiently. Naturally, this provides a huge potential for optimization concerning the quality of service (QoS) such as power consumption, latency, reliability, or even the throughput of the system. This thesis focuses on a more theoretical view and highlights fundamental bounds for distributed sensing, which serve as a basis for developing practical systems.

A current example of a practical system in mobile radio communications are Cloud Radio Access Networks (C-RANs). Here the base stations (BSs) are connected via fronthaul links to a central unit located in the "cloud" [Che+15]. The main advantage lies in the fact that baseband processing is not performed at the base stations anymore, but in the computationally strong central unit allowing joint processing of multiple users. Therefore, the BSs only operate as radio units (RUs). However, these RUs can still perform a limited amount of low computationally complex signal processing tasks like sampling or compression. Due to capacity-limited fronthaul links a previous compression on the RUs is often required and receives attention in the current literature [Par+14].

Another example is defined by distributed sensor networks forwarding their measurements via capacity-limited links to a central processing unit. In general, these sensors are quite cheap and cannot perform complex processing tasks. However, since forward links have capacity constraints, each sensor needs to perform a data compression before forwarding it. This setup is commonly known as distributed source coding. In information theory, distributed source coding has been of interest for decades, and significant progress has been achieved in the past. A survey on distributed source coding can be found in [GVD06]. The authors discuss different sensor networks focusing on the separation of source and channel coding and the question of whether an analog or digital transmission is superior. For limited communication among sensors, digital processing requires exponentially more sensors to achieve the same distortion as analog processing. However, for rich communication among sensors, digital processing works fine.

Considering digital processing only, different assumptions on the sensor architecture can be found in the literature. Multi-terminal source coding generally considers  $M$  correlated

sources and  $N$  decoders being interested in a subset of only  $m \leq M$  source signals. For instance, this could be a network containing two sensors. One is measuring the temperature and the other the pressure. Results for multi-terminal source coding can only be found for specific probability distributions of the source signal as well as specific distortion measures. In [HK80], an inner bound on the achievable rate region has been derived, and [CW14] provide a solution for two encoders under log-loss distortion measure. The quadratic Gaussian multi-terminal source coding problem for two encoders has been solved in [WTV08]. Oohama developed upper and lower bounds for the quadratic Gaussian multi-terminal source coding problem with correlated source signals in [Ooh08; Ooh12]. However, the rate region of the general multi-terminal source coding problem is still unknown for more than two encoders.

This thesis considers a special case of the multi-terminal source coding problem, where all  $M$  sensors are interested in the same source signal. The scenario is widely known as the Chief Executive (Estimation) Officer (CEO) problem [BZV96]. This simplifying assumption allows the exact determination of the complete rate region for specific source distributions and specific distortion measures but arbitrary network sizes. In the context of C-RANs, this scenario can be regarded as a network with a single user being connected to multiple RUs. In particular, this thesis focuses on the offline design of compression devices of each sensor in the network, maximizing the overall spectral efficiency of the network. Therefore, an information-theoretic approach is pursued. Applying the logarithmic loss distortion measure allows a close link between the CEO problem and the so-called information bottleneck (IB) method [TPB99]. The general optimization algorithm adapts the Blahut-Arimoto algorithm [Bla72; Ari72] and sequentially optimizes each sensor, exploiting the quantizer mappings of previously designed sensors by the Wyner-Ziv coding principle [SK21]. However, this approach results in dimensionality problems, especially for larger networks. Therefore, the IB principle is applied to compress the quantizer mappings of previously designed sensors, exploited for Wyner-Ziv coding, allowing the design even for larger network sizes [Ste+21b; Ste+21a]. Finally, this thesis extends the general CEO problem allowing each sensor to partially communicate with each other during runtime, increasing the overall performance of the network even further [SAK22; SK22].

## 1.2 Structure of Thesis

This thesis is structured as follows: Chapter 2 introduces basic information-theoretic fundamentals which are necessary to follow the argumentation within this thesis. Chapter 3 reviews preliminaries on lossy compression, starting with the general rate-distortion theory and the remote sensing problem. Moreover, this chapter introduces the IB approach as a basic optimization concept used within this thesis. Afterward, the scalar remote sensing problem is extended to the distributed case. Here, the CEO problem with a logarithmic loss distortion measure is defined with different theoretical bounds. This chapter ends with the definition of the fully cooperative CEO scenario, where each sensor has access to all measurements in the network. The main contributions of this thesis can be found in Chapter 4 and Chapter 5 where different algorithmic solutions are introduced to solve the

non-cooperative CEO scenario and the partially cooperative CEO scenario, respectively. In particular, Chapter 4 introduces a greedy approach, which sequentially optimizes each sensor exploiting the statistics of previously designed quantizers using Wyner-Ziv coding. This approach is further elaborated to allow the optimization of large networks as well as an incorporation of imperfect forward channels. In Chapter 5, the non-cooperative CEO scenario is extended to allow partial cooperation among sensors during runtime. This includes different inter-sensor communication protocols, i.e., the successive broadcast transmission protocol, the sequential point-to-point transmission protocol, and the two-phase transmission protocol. Finally, Chapter 6 summarizes the main contributions of this thesis.

### 1.3 Notations

The probability mass function (pmf) of the random variable  $\mathcal{X}$ , denoted in calligraphic letters, is given by  $p(\mathcal{X} = x)$ , where  $x$  is a realization of  $\mathcal{X}$  drawn from the set  $\mathbb{X}$  with cardinality  $|\mathbb{X}|$ . In order to simplify the notation, this pmf is denoted as  $p(x)$  throughout this thesis. Vectors are denoted in bold letters,  $\mathbf{y} = [y_1 \dots y_M]^T$ . In this sense, multivariate random variables are represented by boldface calligraphic letters with  $\mathcal{Z}_{< m}$  covering the processes  $\mathcal{Z}_1$  to  $\mathcal{Z}_{m-1}$ . A Markov chain  $\mathcal{X} \rightarrow \mathcal{Y} \rightarrow \mathcal{Z}$  implies that  $p(x, y, z) = p(z|y)p(y|x)p(x)$ , where  $p(y|x)$  is a conditional pmf. The expectation of a function  $f(x)$  with respect to  $x$  is denoted by  $\mathbb{E}_{\mathcal{X}} [f(\mathcal{X})]$ . A univariate Gaussian distribution is given by  $\mathcal{N}(\mu, \sigma^2)$  with mean  $\mu$  and variance  $\sigma^2$ . Similarly, the complex Gaussian distribution is given by  $\mathcal{CN}(\mu, \sigma^2)$ . The entropy of the discrete random variable  $\mathcal{X}$  is denoted by  $H(\mathcal{X})$ . The mutual information between  $\mathcal{X}$  and  $\mathcal{Y}$  is given by  $I(\mathcal{X}; \mathcal{Y})$ . Moreover, divergence measures  $D_{\text{KL}}[\cdot||\cdot]$  and  $D_{\text{JS}}^{\Pi}[\cdot||\cdot]$  denote the Kullback-Leibler divergence and the Jensen-Shannon divergence, respectively.

# Chapter 2

## Fundamentals

This chapter provides the necessary theoretical background in information theory in order to be able to follow the argumentation and understand the main contributions of this thesis. In Section 2.1, general definitions of different probability densities and distributions, as well as some characteristics and mathematical properties, are given. Section 2.2 covers information-theoretic measures like entropy and mutual information based on definitions of Shannon in [Sha48]. Moreover, different ways to compare distributions are introduced by the Kullback-Leibler divergence and the Jensen-Shannon divergence. Finally, Section 2.3 introduces selected channels, and their capacities are discussed.

### 2.1 Densities and Probabilities

#### 2.1.1 Continuous Probability Densities

Let  $\mathcal{X}$  be a continuous random variable with  $x$  being a realization of  $\mathcal{X}$  drawn from an uncountable set  $\mathbb{X}$ . The probability density function (pdf)  $p(x)$  of  $\mathcal{X}$  is defined as a non-negative function which integrates to one [Kol33; GS09; PP02]

$$\int_{\mathbb{X}} p(x) dx = 1 . \quad (2.1)$$

The probability  $\Pr(\tau_1 \leq x \leq \tau_2)$  of a realization falling within a particular range  $\tau_1 \leq x \leq \tau_2$  can be determined from  $p(x)$  with

$$\Pr(\tau_1 \leq x \leq \tau_2) = \int_{\tau_1}^{\tau_2} p(x) dx . \quad (2.2)$$

However, working with continuous random variables requires analytical expressions of the pdf of a random variable, which might not be known. Therefore, all algorithms within this thesis work with discrete probability distributions.

#### 2.1.2 Discrete Probability Distributions

The major difference to the continuous case is that a discrete random variable  $\mathcal{X}$  draws realizations  $x$  from the set  $\mathbb{X}$ , which has a countable event space. Within this thesis, all event spaces of discrete random variables have a finite cardinality, e.g.,  $|\mathbb{X}| < \infty$ . The probability of a specific realization  $x$  is given by  $\Pr\{\mathcal{X} = x\} \geq 0$ . Analog to the continuous case, the sum over all probabilities is equal to one [Kol33; GS09; PP02]

$$\sum_{x \in \mathbb{X}} \Pr\{\mathcal{X} = x\} = 1 . \quad (2.3)$$

A pmf  $\Pr\{\mathcal{X}\}$  of a discrete random variable  $\mathcal{X}$  is defined by the probabilities of all  $x \in \mathbb{X}$ . It can be interpreted as a superposition of Dirac impulses located at the realizations  $x$  and weighted with the corresponding probabilities leading to a generalized pdf definition

$$p(x) = \sum_{x_k \in \mathbb{X}} \Pr\{\mathcal{X} = x_k\} \delta(x - x_k) . \quad (2.4)$$

Therefore, to simplify the notation, the pmf can also be denoted as  $p(x)$ , similar to the continuous case. For brevity, the summation over the event space  $\sum_{x \in \mathbb{X}}$  will be denoted by  $\sum_x$ .

### 2.1.3 Joint Probability Distribution

Let  $\mathcal{X}$  and  $\mathcal{Y}$  be two discrete random variables, the joint probability distribution  $p(x, y)$  represents the probabilities that the realizations  $x \in \mathbb{X}$  and  $y \in \mathbb{Y}$  are observed together for all pairs  $(x, y) \in \mathbb{X} \times \mathbb{Y}$ , where  $\mathbb{X} \times \mathbb{Y} = \{(x, y) | x \in \mathbb{X}, y \in \mathbb{Y}\}$  denotes the Cartesian product. Note the law of total probability states

$$\sum_x \sum_y p(x, y) = 1 . \quad (2.5)$$

The joint probability distribution  $p(x, y)$  can be used to obtain the marginal distributions by  $p(x) = \sum_y p(x, y)$  and  $p(y) = \sum_x p(x, y)$ . This marginalization is referred to as the sum rule of probabilities [Bis06]. In general,  $p(x, y)$  can not be constructed out of the marginals  $p(x)$  and  $p(y)$ . For the special case of  $\mathcal{X}$  and  $\mathcal{Y}$  being independent random variables, it is possible to obtain the joint probability distribution by  $p(x, y) = p(x) \cdot p(y)$ .

### 2.1.4 Conditional Probability Distribution

Let  $\mathcal{X}$  and  $\mathcal{Y}$  be two discrete random variables, the conditional probability distribution  $p(y|x)$  defines the probability of  $y$  given the specific value  $x$ . In other words, it defines the probability of an event after observing another event, which is the basic task in estimation theory. For the conditional probability distribution

$$\sum_y p(y|x) = 1 \quad \forall x \in \mathbb{X} \quad (2.6)$$

holds. In addition, the joint probability distribution can be obtained by means of the conditional probability distribution with

$$p(y, x) = p(y|x) \cdot p(x) \quad (2.7)$$

which is referred to as the product rule of probabilities [Bis06].

### 2.1.5 Expectation and Variance

Let  $f(x)$  define an arbitrary function of a discrete random variable  $\mathcal{X}$ . The expectation of  $f(x)$  is defined as

$$\mathbb{E}_{\mathcal{X}} [f(x)] = \sum_x f(x)p(x). \quad (2.8)$$

With  $f(x) = x$ , it defines the average value  $\mu_x$  of the function  $f(x)$  with respect to the probability distribution  $p(x)$ . For joint distributions, the expectation can be calculated as

$$\mathbb{E}_{\mathcal{X},\mathcal{Y}} [f(x, y)] = \sum_x \sum_y f(x, y)p(x, y). \quad (2.9)$$

Note that the subscripts denote the random variable over which the average is determined. In (2.9), the expectation is calculated with respect to both random variables. For conditional distributions, the expectation can be determined as

$$\mathbb{E}_{\mathcal{X}|y} [f(x)] = \sum_x f(x)p(x|y). \quad (2.10)$$

Finally, the variance of a discrete random variable  $\mathcal{X}$  can be calculated by

$$\text{Var}(\mathcal{X}) = \mathbb{E}_{\mathcal{X}} [(\mathcal{X} - \mathbb{E}_{\mathcal{X}} [\mathcal{X}])^2] = \mathbb{E}_{\mathcal{X}} [\mathcal{X}^2] - \mathbb{E}_{\mathcal{X}} [\mathcal{X}]^2. \quad (2.11)$$

## 2.2 Information Theory

### 2.2.1 Information and Entropy

Let  $\mathcal{X}$  be a discrete random variable with pmf  $p(x)$ , the amount of information belonging to a specific event  $x \in \mathbb{X}$  is defined as

$$H(\mathcal{X} = x) = -\log_b p(x) \quad (2.12)$$

where the base  $b$  defines the unit of the information [CT06]. In communications, the most common base is  $b = 2$  leading to the unit [bit]. The information  $H(\mathcal{X} = x)$  is a non-negative and a monotonically decreasing function in  $p(x)$  depending on probabilities only. The more likely a specific event, the less its amount of information. Hence, the information of a reliable event is zero, while the event with probability zero has infinite information. The average amount of information defines the entropy  $H(\mathcal{X})$  and is given by

$$H(\mathcal{X}) = \mathbb{E}_{\mathcal{X}} [-\log_2 p(x)] = -\sum_x p(x) \log_2 p(x). \quad (2.13)$$



The entropy  $H(\mathcal{X})$  is a measure of uncertainty. Therefore, the maximum entropy is obtained by a uniform distribution of  $\mathcal{X}$  with  $p(x) = \frac{1}{|\mathbb{X}|}$  leading to  $H(X) = \log_2 |\mathbb{X}|$ . For a second random variable  $\mathcal{Y}$ , the entropy extends to the joint entropy

$$H(\mathcal{X}, \mathcal{Y}) = \mathbb{E}_{\mathcal{X}, \mathcal{Y}} [-\log_2 p(x, y)] = \sum_x \sum_y p(x, y) \log_2 p(x, y). \quad (2.14)$$

If  $\mathcal{X}$  and  $\mathcal{Y}$  are independent, the joint entropy is achieved by the sum of the individual entropies  $H(\mathcal{X}, \mathcal{Y}) = H(\mathcal{X}) + H(\mathcal{Y})$ . According to the chain rule for entropies, this joint entropy can be rewritten as  $H(\mathcal{X}, \mathcal{Y}) = H(\mathcal{X}) + H(\mathcal{Y}|\mathcal{X})$  or as  $H(\mathcal{X}, \mathcal{Y}) = H(\mathcal{Y}) + H(\mathcal{X}|\mathcal{Y})$ , where  $H(\mathcal{Y}|\mathcal{X})$  defines the conditional entropy of  $\mathcal{Y}$  given  $\mathcal{X}$  and  $H(\mathcal{X}|\mathcal{Y})$  defines the conditional entropy of  $\mathcal{X}$  given  $\mathcal{Y}$ .

$$H(\mathcal{Y}|\mathcal{X}) = \mathbb{E}_{\mathcal{X}, \mathcal{Y}} [-\log_2 p(y|x)] = - \sum_x \sum_y p(x, y) \log_2 p(y|x) \quad (2.15)$$

Note that if  $\mathcal{X}$  and  $\mathcal{Y}$  are correlated, the conditional entropy is always smaller than the unconditional entropy  $H(\mathcal{X}|\mathcal{Y}) < H(\mathcal{X})$  since more information always decreases the uncertainty about  $\mathcal{X}$ . However, if  $\mathcal{X}$  and  $\mathcal{Y}$  are uncorrelated, the conditional entropy equals the unconditional entropy  $H(\mathcal{X}|\mathcal{Y}) = H(\mathcal{X})$ , since  $\mathcal{Y}$  contains no information about  $\mathcal{X}$ . In the case of  $\mathcal{X}$  being a continuous random variable with a pdf  $p(x)$  the definition of entropy can be adapted to

$$h(\mathcal{X}) = - \int_{\mathbb{S}} p(x) \log_2 p(x) dx \quad (2.16)$$

where  $h(\mathcal{X})$  is termed the differential entropy and  $\mathbb{S}$  defines the support of  $\mathcal{X}$ , i.e. the set where  $p(x) > 0$ . Note that the differential entropy can become negative. Therefore, an interpretation in the sense of a measure of uncertainty becomes a bit vague. As in the discrete case, the differential entropy can easily be extended to the multivariate and conditional case.

**Entropy coding:** The information-theoretic measure entropy finds application, e.g., in lossless data compression. Given a source process  $\mathcal{X}$  which delivers a sequence  $\mathbf{x} = \{x_1, x_2, \dots, x_n\}$  with  $n \rightarrow \infty$ , Shannon's source coding theorem [Sha48] defines the entropy  $H(\mathcal{X})$  as the ultimate lower bound for lossless compression, motivating the name *entropy coding*. That means representing the whole sequence requires at least  $nH(\mathcal{X})$  bit. A very efficient and optimal algorithm for this source coding problem is the Huffman coding, first published in [Huf52]. Applying this coding strategy results in a prefix-free code. Hence, the bit string representing a particular symbol is never a prefix of the bit string representing any other symbol.

## 2.2.2 Mutual Information

Based on [CT06], the mutual information  $I(\mathcal{X}; \mathcal{Y})$  is defined as the common information of the two random variables  $\mathcal{X}$  and  $\mathcal{Y}$ . It is a positive and symmetric measure  $I(\mathcal{X}; \mathcal{Y}) =$

$I(\mathcal{Y}; \mathcal{X}) \geq 0$  bit/s/Hz. Naturally, if  $\mathcal{X}$  and  $\mathcal{Y}$  are independent the mutual information becomes zero, i.e.  $I(\mathcal{X}; \mathcal{Y}) = 0$  bit/s/Hz.

In the case of  $\mathcal{X}$  and  $\mathcal{Y}$  being discrete random variables, it can be calculated as

$$I(\mathcal{X}; \mathcal{Y}) = \underbrace{\mathbb{E}_{\mathcal{X}, \mathcal{Y}} \left[ \log_2 \frac{p(x, y)}{p(x) \cdot p(y)} \right]}_{H(\mathcal{X}) + H(\mathcal{Y}) - H(\mathcal{X}, \mathcal{Y})} = \underbrace{\mathbb{E}_{\mathcal{X}, \mathcal{Y}} \left[ \log_2 \frac{p(x|y)}{p(x)} \right]}_{H(\mathcal{X}) - H(\mathcal{X}|\mathcal{Y})} = \underbrace{\mathbb{E}_{\mathcal{X}, \mathcal{Y}} \left[ \log_2 \frac{p(y|x)}{p(y)} \right]}_{H(\mathcal{Y}) - H(\mathcal{Y}|\mathcal{X})} \quad (2.17)$$

$$= \sum_x \sum_y p(x, y) \log_2 \frac{p(x|y)}{p(x)} = \sum_x \sum_y p(x, y) \log_2 \frac{p(y|x)}{p(y)}. \quad (2.18)$$

The mutual information can be interpreted as the reduction of uncertainty about  $\mathcal{X}$  by observing  $\mathcal{Y}$  expressed by  $H(\mathcal{X}) - H(\mathcal{X}|\mathcal{Y})$  or the uncertainty remaining about  $\mathcal{Y}$  when knowing  $\mathcal{X}$  expressed by  $H(\mathcal{Y}) - H(\mathcal{Y}|\mathcal{X})$ . For continuous random variables, the entropies  $H(\cdot)$  in (2.17) become differential entropies  $h(\cdot)$ , which simply results in an exchange of the sums by integrals in (2.18). Although the differential entropies can become negative, the interpretation of the mutual information is the same as for discrete random variables. Note that the unit of the mutual information is [bit/s/Hz] and describes the ratio between the data rate and the bandwidth. This ratio is also known as the spectral efficiency. Within this thesis, mutual information and spectral efficiency are used interchangeably. The conditional mutual information  $I(\mathcal{X}; \mathcal{Y}|\mathcal{Z})$  describes the common information between the two random variables  $\mathcal{X}$  and  $\mathcal{Y}$  given another random variable  $\mathcal{Z}$ . It can be defined as

$$I(\mathcal{X}; \mathcal{Y}|\mathcal{Z}) = \mathbb{E}_{\mathcal{X}, \mathcal{Y}, \mathcal{Z}} \left[ \log_2 \frac{p(x, y|z)}{p(x|z) \cdot p(y|z)} \right] = \mathbb{E}_{\mathcal{X}, \mathcal{Y}, \mathcal{Z}} \left[ \log_2 \frac{p(y|x, z)}{p(y|z)} \right]. \quad (2.19)$$

The additional information  $\mathcal{Z}$  can either increase or decrease the mutual information  $I(\mathcal{X}; \mathcal{Y}|\mathcal{Z})$  compared to the unconditioned case  $I(\mathcal{X}; \mathcal{Y})$ . If  $\mathcal{Z}$  contains information about the disturbance in  $\mathcal{Y}$ , the knowledge of  $\mathcal{Z}$  increases the conditional mutual information. However, if  $\mathcal{Z}$  contains information about the random variable  $\mathcal{Y}$ , e.g., if  $\mathcal{X} \rightarrow \mathcal{Y} \rightarrow \mathcal{Z}$  form a Markov chain, additional information  $\mathcal{Z}$  will always decrease the mutual information  $I(\mathcal{X}; \mathcal{Y}|\mathcal{Z}) \leq I(\mathcal{X}; \mathcal{Y})$ .

The mutual information can easily be extended to multivariate random variables. Let  $\mathcal{X}$  be a univariate random variable and  $\mathbf{Y} = \mathcal{Y}_{1:M}$  be a multivariate random variable covering the processes  $\mathcal{Y}_1, \dots, \mathcal{Y}_M$ . The mutual information between  $\mathcal{X}$  and  $\mathbf{Y}$  is defined as

$$I(\mathcal{X}; \mathbf{Y}) = \mathbb{E}_{\mathcal{X}, \mathbf{Y}} \left[ \log_2 \frac{p(x, y_1, \dots, y_M)}{p(x) \cdot p(y_1, \dots, y_M)} \right]. \quad (2.20)$$

This multivariate mutual information can be decoupled into a sum of conditional mutual information using the chain rule for mutual information

$$I(\mathcal{X}; \mathbf{Y}) = \sum_{i=1}^M I(\mathcal{X}; \mathcal{Y}_i | \mathcal{Y}_1, \dots, \mathcal{Y}_{i-1}). \quad (2.21)$$

Note that the order of decoupling, i.e., the order of decoding, can be chosen arbitrarily and does not affect the overall mutual information.

**Data processing inequality:** The data processing inequality is a basic theorem in information theory. Given the Markov chain  $\mathcal{X} \rightarrow \mathcal{Y} \rightarrow \mathcal{Z}$ , i.e.,  $p(x, y, z) = p(x)p(y|x)p(z|y)$ , the data processing inequality is given by

$$I(\mathcal{X}; \mathcal{Y}) \geq I(\mathcal{X}; \mathcal{Z}). \quad (2.22)$$

In other words, without adding new information, the uncertainty about  $\mathcal{X}$  given the observation  $\mathcal{Y}$  can not be further decreased by any processing of  $\mathcal{Y}$  to  $\mathcal{Z}$ . Therefore, lost information can not be recovered without a further source of information.

### 2.2.3 Kullback-Leibler divergence

The Kullback-Leibler (KL) divergence, also called relative entropy, is a statistical measure of similarity between two random distributions. Given the two distributions  $p_1(x)$  and  $p_2(x)$  the KL divergence is defined as

$$D_{\text{KL}} [p_1||p_2] = \mathbb{E}_{p_1(x)} \left[ \log_b \frac{p_1(x)}{p_2(x)} \right] = \sum_x p_1(x) \log_b \frac{p_1(x)}{p_2(x)}. \quad (2.23)$$

The base  $b$  defines the unit in which the KL divergence is measured. The KL divergence represents the expected error assuming the distribution  $p_2(x)$  while the true distribution is  $p_1(x)$ . From the above equation, it is easy to conclude that for  $p_1(x) = p_2(x)$ , the KL divergence is zero. In addition, the support of  $p_1(x)$  must be a subset of the support of  $p_2(x)$ , which means that if there exists any event  $x$  in  $p_1(x)$  which is not included or zero in  $p_2(x)$  the  $D_{\text{KL}} [p_1||p_2] \rightarrow \infty$ , hence it is unbounded. Moreover, it can be shown that the KL divergence is always non-negative [CT06]. It has to be mentioned that the KL divergence is not a true distance since it is not symmetric, i.e.,  $D_{\text{KL}} [p_1||p_2] \neq D_{\text{KL}} [p_2||p_1]$ , and the triangle inequality is not satisfied. Note that the mutual information  $I(\mathcal{X}; \mathcal{Y})$  can be expressed by a KL divergence between  $p(x, y)$  and  $p(x)p(y)$  as

$$I(\mathcal{X}; \mathcal{Y}) = D_{\text{KL}} [p(x, y)||p(x)p(y)] = \sum_{x, y} p(x, y) \log_2 \frac{p(x, y)}{p(x)p(y)}. \quad (2.24)$$

### 2.2.4 Jensen-Shannon divergence

Since the KL divergence is not symmetric and unbounded, it might be convenient to introduce another statistical distance measure. Based on the previously defined KL divergence for two probability distributions  $p_1(x)$  and  $p_2(x)$  the Jensen-Shannon (JS) divergence [Lin91] is given by

$$D_{\text{JS}}^{\Pi} [p_1||p_2] = \pi_1 D_{\text{KL}} [p_1||\bar{p}] + \pi_2 D_{\text{KL}} [p_2||\bar{p}] \quad \text{with } \bar{p} = \pi_1 p_1 + \pi_2 p_2, \quad (2.25)$$

where  $\Pi = \{\pi_1, \pi_2\}$ ,  $0 < \pi_1, \pi_2 < 1$  and  $\pi_1 + \pi_2 = 1$ . This extension to the KL divergence has some notable advantages over the original one. While the KL divergence is unbounded, i.e., it can become infinity, the JS divergence is always bounded to a finite value  $0 \leq D_{\text{JS}}^{\Pi} [p_1||p_2] \leq \log_b 2$ . Again,  $b$  defines the unit in which the JS divergence is measured.

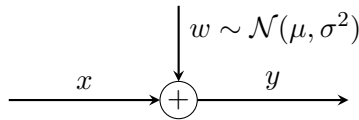


Figure 2.1: Schematic representation of an AWGN channel

Moreover, the JS divergence is a symmetric measure, i.e.,  $D_{\text{JS}}^{\Pi} [p_1 || p_2] = D_{\text{JS}}^{\Pi} [p_2 || p_1]$ . While the KL divergence is very sensitive for low probabilities in the second argument  $p_2$ , the JS divergence is more robust in this case. However, since the triangle inequality is not fulfilled, it is still not a true distance.

## 2.3 Channel Models

### 2.3.1 Channel Capacity

Casually stated, the channel capacity defines the maximum amount of information that can be transmitted over a given channel. In his seminal work, "A Mathematical Theory of Communication" from 1948, Shannon defined the theoretical bounds for reliable communication over an arbitrary channel [Sha48]. Given a channel with input  $\mathcal{X}$  and output  $\mathcal{Y}$ , its capacity is defined as the supremum of the mutual information  $I(\mathcal{X}; \mathcal{Y})$  with respect to the input distribution  $p(x)$ .

$$C = \sup_{p(x)} I(\mathcal{X}; \mathcal{Y}) \quad (2.26)$$

### 2.3.2 Additive White Gaussian Noise Channel

The additive white Gaussian noise (AWGN) channel is one of the most fundamental channel models used in communications for point-to-point transmission. Let  $x$  be the input of an AWGN channel. The output is given by  $y = x + w$ , where  $w$  denotes the additive noise, as shown in Figure 2.1. The name AWGN already implies that  $w$  is white and Gaussian distributed with  $w \sim \mathcal{CN}(0, \sigma_w^2)$ . The Gaussian distribution is motivated by the central limit theorem, stating that the superposition of independent disturbances results in a Gaussian distribution. Moreover, white noise has a constant power spectral density. A common way to quantify this channel model is the signal-to-noise ratio (SNR)  $\gamma = \frac{\sigma_x^2}{\sigma_w^2}$ , where  $\sigma_x^2$  and  $\sigma_w^2$  denote the signal power and the noise variance, respectively.

**Channel Capacity:** Given the general definition of the channel capacity in (2.26) and the definition of the mutual information in (2.17), the capacity for an AWGN channel can be derived in closed form. Since the mutual information is defined as  $I(\mathcal{X}; \mathcal{Y}) = h(y) - h(y|x)$  with  $p(y|x)$  being the fixed channel likelihood, it is obvious that  $h(y)$  has to become maximum in order to maximize the mutual information. As this differential entropy becomes maximal for a Gaussian distribution  $y \sim \mathcal{CN}(0, \sigma_y^2)$  being derived by the

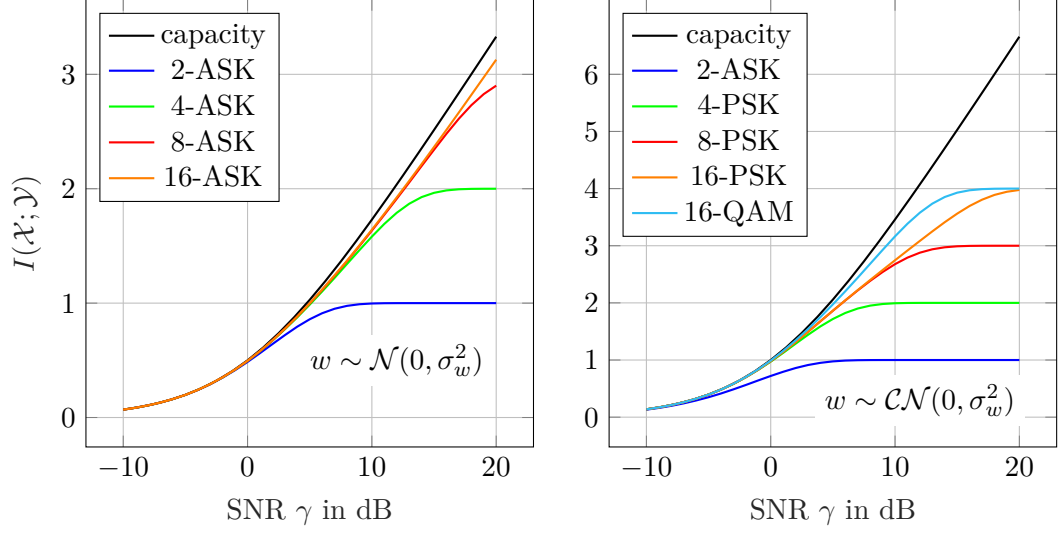


Figure 2.2: Capacity and spectral efficiency of an AWGN channel for different digital modulation schemes

convolution  $p(y) = p(x) * p(w)$  with  $w \sim \mathcal{CN}(0, \sigma_w^2)$ , it is clear that the capacity is achieved for  $x \sim \mathcal{CN}(0, \sigma_x^2)$ .

$$\begin{aligned}
 C &= \sup_{p(x)} I(\mathcal{X}; \mathcal{Y}) \\
 &= \sup_{p(x)} h(\mathcal{Y}) - h(\mathcal{Y}|\mathcal{X}) \\
 &= \log_2(\pi e \sigma_y^2) - \log_2(\pi e \sigma_w^2) \\
 &= \log_2\left(1 + \frac{\sigma_x^2}{\sigma_w^2}\right) \\
 &= \log_2(1 + \gamma)
 \end{aligned} \tag{2.27}$$

In the case that  $x$  and  $w$  are real-valued, i.e.,  $x \sim \mathcal{N}(0, \sigma_x^2)$  and  $w \sim \mathcal{N}(0, \sigma_w^2)$ , the capacity of the AWGN channel is given by

$$C = \frac{1}{2} \log_2(1 + \gamma) \tag{2.28}$$

The channel capacity for an AWGN channel is just dependent on the signal-to-noise ratio  $\gamma = \frac{\sigma_x^2}{\sigma_w^2}$ .

Figure 2.2 depicts the channel capacity for a real-valued AWGN channel and a complex AWGN channel as well as the spectral efficiency using different modulation schemes. The black curve represents the capacity achieved by a Gaussian input distribution  $p(x)$ . Naturally, using discrete input distributions for discrete digital modulation schemes, the spectral efficiency is upper bounded by  $\log_2 M$  with  $M$  being the modulation order independent of  $\sigma_x^2$ . For lower SNR values, the spectral efficiency using discrete input values does not differ from the capacity. For higher-order mappings, the bifurcation point from the capacity is located at higher SNRs.

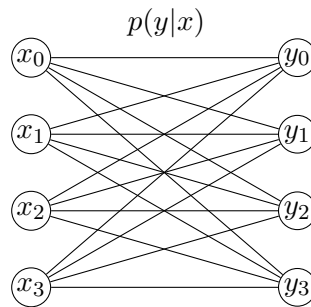


Figure 2.3: Schematic representation of an DMC channel

### 2.3.3 Discrete Memoryless Channel

In general, a discrete memoryless channel (DMC) is an abstract model where the channel has a discrete number of inputs connected to a discrete number of outputs. Figure 2.3 shows an example of a DMC for an input  $\mathcal{X}$  with cardinality of  $|\mathbb{X}| = 4$  and an output  $\mathcal{Y}$  with cardinality  $|\mathbb{Y}| = 4$ . Each input is connected to each output with a corresponding transition probability. The DMC is completely defined by all transition probabilities  $p(y|x)$ , which can be stored as a transition matrix  $\mathbf{P}$  with rows and columns representing realizations of  $\mathcal{Y}$  and  $\mathcal{X}$ , respectively.

**Channel Capacity:** The capacity of a DMC can be determined using the general definition of the capacity in (2.26) and the definition of the mutual information for discrete random variables in (2.17). It is easy to see that a symmetric DMC with  $\mathbf{P} = \mathbf{P}^T$ , combined with a uniform distributed input with  $p(x) = \frac{1}{|\mathbb{X}|}$  results in a uniform distributed output  $p(y) = \frac{1}{|\mathbb{Y}|}$ . Since a uniform distribution maximizes the entropy  $H(\mathcal{Y})$ , this delivers the capacity of the DMC. In this case, it can be computed as

$$\begin{aligned}
 C &= \sup_{p(x)} I(\mathcal{X}; \mathcal{Y}) \\
 &= \sup_{p(x)} H(\mathcal{Y}) - H(\mathcal{Y}|\mathcal{X}) \\
 &= H(\mathcal{Y}) - H(\mathbf{r}) \\
 &= \log_2(|\mathbb{Y}|) - H(\mathbf{r}) ,
 \end{aligned} \tag{2.29}$$

where  $\mathbf{r}$  defines a specific row (or column) of the transition matrix  $p(y|x)$  [CT06].

## 2.4 Discussion

This chapter introduced the necessary theoretical background in information theory. General definitions of probability densities and distributions, as well as some properties and characteristics, have been defined. Afterward, Shannon's basic definitions like entropy and mutual information were discussed, followed by definitions of Kullback-Leibler divergence and Jensen-Shannon divergence. Finally, basic channel models used in this thesis

have been introduced. With this background information, the reader has the necessary knowledge to understand the preliminaries on lossy compression in the next chapter.

## Chapter 3

### Preliminaries on Lossy Compression

This chapter introduces basic concepts of lossy data compression. Starting with the scalar case, Section 3.1 revisits Shannon’s famous rate-distortion theory [Sha48; Sha59]. Lossy compression with some side-information is defined by the Wyner-Ziv coding principle [Wyn78; WZ76] in Section 3.2. The rate-distortion theory for clean sources is extended to noisy sources, defining the remote sensing problem in Section 3.3. In Section 3.4, the IB principle [TPB99] is defined, as well as some connections to the remote source coding problem. Moreover, the iterative IB algorithm is introduced as an algorithmic approach to solve the IB problem. Section 3.5 considers a special case of the IB approach for erroneous channels between encoder and receiver based on the work of Winkelbauer in [Win14; WMB13]. The scalar remote sensing scenario is extended to the distributed case in Section 3.6. Subsection 3.6.1 defines the non-cooperative remote sensing scenario, where the communication among sensors is not possible. This is widely known as the CEO problem. The general system model is introduced, as well as theoretical bounds for the rate region. A lower bound for algorithms introduced in Chapter 4 is given as independent IB optimization. Finally, Subsection 3.6.2 introduces the fully cooperative remote sensing scenario and an algorithmic approach to solve it. The results of this scenario deal as an upper bound for algorithms introduced in subsequent chapters.

#### 3.1 Rate-Distortion Theory

As described in Section 2.2, the lower bound for lossless compression is the entropy. Compressing below this bound yields an inevitable loss of information. The general idea of rate-distortion theory goes back to the work of Shannon [Sha48; Sha59]. It defines the theoretical foundations for this lossy data compression. The first group working on this topic after Shannon’s seminal publications studied at Moscow University, including A. N. Kolmogorov. They made contributions on a topic called  $\epsilon$ -entropy, today known as the rate-distortion theory [Kol56]. Later in the seventies, amongst others, the group around T. Berger [Ber71][Ber75], the group around R. M. Gray [Gra90], and the group around A.D. Wyner contributed to the rate-distortion theory with [WZ71] defining theoretical bounds on the rate-distortion function where they considered stationary sources with memory.

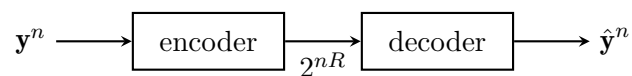


Figure 3.1: Schematic illustration of lossy compression [EK11]



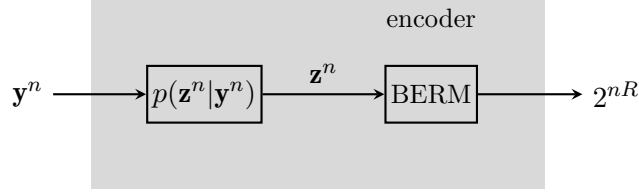


Figure 3.2: Encoding process for lossy source coding

Later work considers the rate-distortion problem with side-information [WW75; WZ76; Wyn78].

For more background information on the historical development of the rate-distortion theory, the reader might refer to the overview paper of T. Berger [BG98], which provides a detailed description of the main progression of different research groups in this topic for the last sixty years. In addition, the survey of J. C. Kieffer [Kie93] provides a good overview concerning source coding with a fidelity criterion.

A general model for the rate-distortion setup is given in Figure 3.1. The encoding process is a function

$$f_n : \mathcal{Y}^n \rightarrow \{1, 2, \dots, 2^{nR}\} \quad (3.1)$$

which represents the source sequence  $\mathbf{y}^n = [y[1], \dots, y[n]]^T$  of length  $n$  using  $nR$  bits. The encoded bits shall be transmitted over a perfect channel to a decoder. Equivalently, the decoding process is a function

$$g_n : \{1, 2, \dots, 2^{nR}\} \rightarrow \hat{\mathcal{Y}}^n \quad (3.2)$$

which tries to reconstruct the source sequence. The average distortion between  $\mathbf{y}^n$  and the reconstructed values  $\hat{\mathbf{y}}^n$  is defined by

$$\mathbb{E}_{\mathcal{Y}^n, \hat{\mathcal{Y}}^n} [\tilde{d}^n(\mathbf{y}^n, \hat{\mathbf{y}}^n)] = \mathbb{E}_{\mathcal{Y}^n, \hat{\mathcal{Y}}^n} \left[ \frac{1}{n} \sum_{i=1}^n \tilde{d}(y_i, \hat{y}_i) \right] \quad (3.3)$$

with  $\tilde{d} : \mathcal{Y} \times \hat{\mathcal{Y}} \rightarrow \mathbb{R}^+$  defining an arbitrary distortion measure. In general, a trade-off between the average distortion and a particular compression measured by the mutual information  $I(\mathcal{Y}^n; \hat{\mathcal{Y}}^n)$  has to be determined. The optimization problem can be defined as

$$R(D) = \min_{p(\hat{\mathbf{y}}^n | \mathbf{y}^n) : \mathbb{E}_{\mathcal{Y}^n, \hat{\mathcal{Y}}^n} \left[ \frac{1}{n} \sum_{i=1}^n \tilde{d}(y_i, \hat{y}_i) \right] \leq D} I(\mathcal{Y}^n; \hat{\mathcal{Y}}^n), \quad (3.4)$$

which tries to find a mapping  $p(\hat{\mathbf{y}}^n | \mathbf{y}^n)$  that minimizes the mutual information  $I(\mathcal{Y}^n; \hat{\mathcal{Y}}^n)$ , i.e., it maximizes the compression, while the average distortion remains below a certain tolerated threshold  $D$ . The encoder in Figure 3.1 can be further defined as in Figure 3.2. It contains a compression step realized by a vector quantizer  $p(\mathbf{z}^n | \mathbf{y}^n)$  and a transformation into a bit stream. This bit stream transformation also incorporates a rate-matching mechanism depending on the type of quantizer mapping in order to ensure a length of

$nR$  bits. It is termed binary encoding with rate-matching (BERM). The vector quantizer is defined by the mapping  $p(\mathbf{z}^n|\mathbf{y}^n)$  which compresses the sequence  $\mathbf{y}^n$  to  $\mathbf{z}^n$ . The mutual information  $I(\mathcal{Y}^n; \mathcal{Z}^n)$  is named *compression rate* and is often interpreted as the rate with which the information is transmitted. However, this only holds for the special case, where the mapping  $p(\mathbf{z}^n|\mathbf{y}^n)$  is deterministic. In this case, the compression rate equals the entropy of the quantizer output  $\mathbf{z}^n$ , i.e.,  $I(\mathcal{Y}^n; \mathcal{Z}^n) = H(\mathcal{Z}^n) = nR$  holds. Therefore, the BERM block only contains lossless entropy coding. For stochastic mappings the compression rate is given by  $I(\mathcal{Y}^n; \mathcal{Z}^n) = H(\mathcal{Z}^n) - H(\mathcal{Z}^n|\mathcal{Y}^n)$  with  $H(\mathcal{Z}^n|\mathcal{Y}^n) > 0$ . Hence,  $I(\mathcal{Y}^n; \mathcal{Z}^n) + H(\mathcal{Z}^n|\mathcal{Y}^n) = H(\mathcal{Z}^n) = nR$  requiring a second compression step of the quantizer output  $\mathbf{z}^n$  to mitigate the influence of  $H(\mathcal{Z}^n|\mathcal{Y}^n)$ . In practice, this second compression can be done by random linear binning, so-called hashing, using, e.g., a linear encoder [EK11]. Note that within this thesis, the focus only lies on optimizing the first compression part, i.e., the quantizer.

If we assume the variables in  $\mathbf{y}^n$  to be uncorrelated, a symbol-wise compression can be applied without any loss compared to the vector compression. For a symbol-wise compression, the optimization problem can be rewritten as

$$R(D) = \min_{p(\hat{y}|y): \mathbb{E}_{\mathcal{Y}, \hat{\mathcal{Y}}}[\tilde{d}(y, \hat{y})] \leq D} I(\mathcal{Y}; \hat{\mathcal{Y}}), \quad (3.5)$$

which tries to find a mapping  $p(\hat{y}|y)$  that minimizes the mutual information  $I(\mathcal{Y}; \hat{\mathcal{Y}})$ . Hence, it maximizes the compression while the average distortion remains below a certain tolerated threshold  $D$ . In the following, the vector notation will be omitted, and only symbol-wise compression is considered.

### 3.1.1 Rate-Distortion Function and Distortion-Rate Function

Since the reconstruction function  $g_n$  in (3.2) is bijective for  $\epsilon$ -typical sequences [EK11], the mutual information  $I(\mathcal{Y}; \hat{\mathcal{Y}})$  equals  $I(\mathcal{Y}; \mathcal{Z})$  with  $\mathcal{Z}$  being the compression random variable. Therefore, the rate-distortion function in (3.5) can be rewritten to

$$R(D) = \min_{p(z|y): \mathbb{E}_{\mathcal{Y}, \mathcal{Z}}[\tilde{d}(y, z)] \leq D} I(\mathcal{Y}; \mathcal{Z}). \quad (3.6)$$

This rate-distortion function defines the maximum achievable compression, i.e., the minimum of the compression rate  $I(\mathcal{Y}; \mathcal{Z})$  for a given average distortion  $\mathbb{E}_{\mathcal{Y}, \mathcal{Z}}[\tilde{d}(y, z)] \leq D$ . The optimization is done with respect to the mapping  $p(z|y)$ , where  $\tilde{d}: \mathcal{Y} \times \mathcal{Z} \rightarrow \mathbb{R}^+$  and  $|\mathcal{Z}| \leq |\mathcal{Y}|$ . The mapping  $p(z|y)$  describes a quantization, i.e., compression of the random variable  $\mathcal{Y}$  to  $\mathcal{Z}$ , and can be of a stochastic or a deterministic nature. This mapping has to be adapted in order to minimize the compression rate  $I(\mathcal{Y}; \mathcal{Z})$  while not exceeding a maximum tolerated average distortion  $D$ . Naturally, larger values of  $D$  imply that a stronger compression can be applied. As an extreme case, all elements of  $\mathcal{Y}$  are mapped to just a single element of  $\mathcal{Z}$  which leads to the maximum compression and lowest compression rate  $I(\mathcal{Y}; \mathcal{Z}) = 0$ . The other extreme case is achieved if  $|\mathcal{Y}| = |\mathcal{Z}|$ . In this case, the compression rate can become  $I(\mathcal{Y}; \mathcal{Z}) = H(\mathcal{Y})$ , if a one-to-one mapping is applied. The

distortion  $\tilde{d}(y, z)$  can be measured by different means. It measures the “distance” between the random variable  $\mathcal{Y}$  and its new representation  $\mathcal{Z}$ . Some examples are the Hamming distance  $\tilde{d}(y, z) = d_H(y, \hat{y}(z))$  or the squared Euclidean distance  $\tilde{d}(y, z) = |\hat{y}(z) - y|^2$ , whose expectation results in the mean squared error (MSE). Note, that since  $z$  is in general only a categorical variable like a cluster index, these measures require  $\hat{y}(z)$  to be a physical representation of  $z$ . However, there are also distortion measures like the logarithmic loss function  $\tilde{d}(y, z) = -\log p(y|z)$ , which is based on statistics and does not require physical representations  $\hat{y}(z)$ . Unfortunately, it is not obvious a priori which distortion measure is the one to apply since the definition is application-specific. Due to convenient mathematical properties, the squared Euclidean distance is often used in the context of Gaussian distributions. However, if physical representations are unavailable, it might be preferable to use a purely statistical measure like the logarithmic loss function. The minimization problem in (3.6) can be reformulated using the method of Lagrangian multipliers

$$L_{\text{RD}}[p(z|y)] = I(\mathcal{Y}; \mathcal{Z}) + \beta \mathbb{E}_{\mathcal{Y}, \mathcal{Z}} [\tilde{d}(y, z)]. \quad (3.7)$$

The Lagrange multiplier  $\beta$  can be interpreted as a trade-off parameter between the compression rate and the average distortion. For  $\beta = 0$ , the focus solely lies on minimizing the compression rate, whereas  $\beta \rightarrow \infty$  focuses only on minimizing the average distortion. Figure 3.3 shows the rate-distortion function  $R(D)$  for a Gaussian source  $y \sim \mathcal{N}(0, 1)$  and the squared Euclidean distance distortion measure  $\tilde{d}(y, z) = |\hat{y}(z) - y|^2$ . The blue curve represents the case where the average distortion equals the maximum tolerated average distortion  $D$ , i.e.,  $\mathbb{E}_{\mathcal{Y}, \mathcal{Z}} [\tilde{d}(y, z)] = D$ . The region above the curve is defined as the achievable region where  $\mathbb{E}_{\mathcal{Y}, \mathcal{Z}} [\tilde{d}(y, z)] < D$  or  $I(\mathcal{Y}; \mathcal{Z}) > R(D)$  holds. Each point in this region represents a specific trade-off between the compression rate and the average distortion. In contrast, the region below the curve can not be achieved, since here the compression rate would yield an average distortion  $\mathbb{E}_{\mathcal{Y}, \mathcal{Z}} [\tilde{d}(y, z)] > D$ . The different points of  $R(D)$  are achieved by varying the trade-off parameter  $\beta$ .

If instead of the average distortion  $D$ , a target compression rate  $R$  is given, the distortion-rate function  $D(R)$  can be applied. It defines the minimal achievable average distortion  $\mathbb{E}_{\mathcal{Y}, \mathcal{Z}} [\tilde{d}(y, z)]$  for a given compression rate  $I(\mathcal{Y}; \mathcal{Z}) \leq R$

$$D(R) = \min_{p(z|y): I(\mathcal{Y}; \mathcal{Z}) \leq R} \mathbb{E}_{\mathcal{Y}, \mathcal{Z}} [\tilde{d}(y, z)]. \quad (3.8)$$

Using the method of Lagrangian multipliers (3.8) can be formulated to

$$L_{\text{DR}}[p(z|y)] = \mathbb{E}_{\mathcal{Y}, \mathcal{Z}} [\tilde{d}(y, z)] + \beta I(\mathcal{Y}; \mathcal{Z}), \quad (3.9)$$

which has to be minimized. In this case,  $\beta = 0$  focuses solely on minimizing the average distortion, whereas  $\beta \rightarrow \infty$  focuses only on minimizing the compression rate.

Obviously, the choice of the distortion measure is essential for rate-distortion theory. More precisely, since  $\tilde{d}(y, z)$  is part of the optimization problem in (3.6) and (3.8) this choice

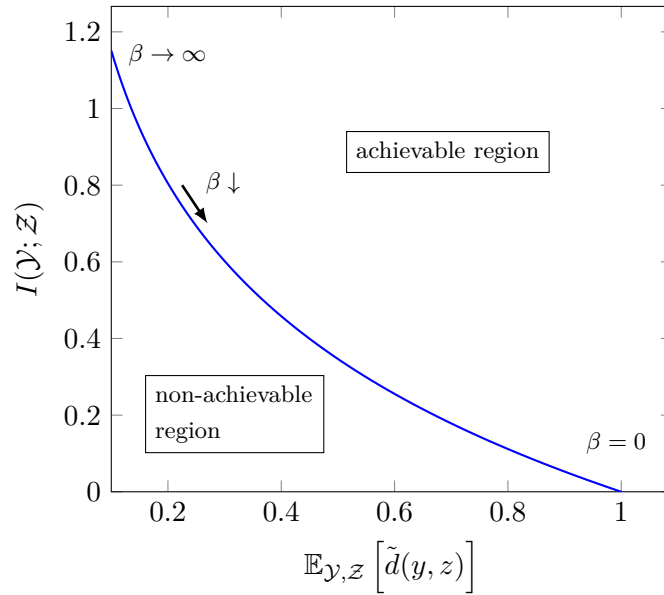


Figure 3.3: Rate-distortion function  $R(D)$  for a Gaussian source  $y \sim \mathcal{N}(0, 1)$  and squared Euclidean distance distortion measure  $\tilde{d}(y, z) = |\hat{y}(z) - y|^2$  showing achievable and non-achievable regions [CT06]

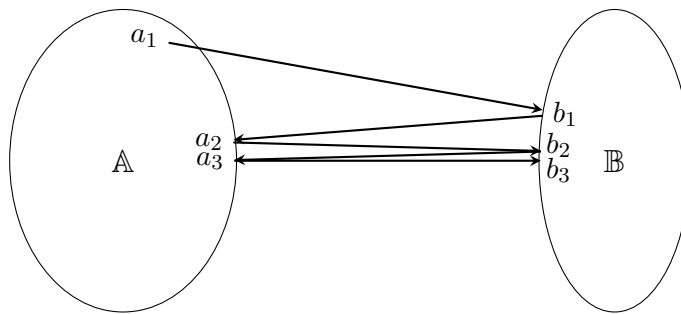


Figure 3.4: Illustration of the convex optimization process applying the Blahut-Arimoto algorithm in  $\mathbb{R}^2$  [CT06; Slo02]

influences its mathematical structure, i.e., whether it is convex or not. If the mapping  $p(z|y)$  is not part of  $\tilde{d}(y, z)$ , the optimization problem is convex [CT06]. This is important since minimizing over a convex function results in the global optimum rather than just a local optimum if it is not convex.

### 3.1.2 Blahut-Arimoto Algorithm

The Lagrangian in (3.9) can be minimized by calculating the derivative with respect to the mapping  $p(z|y)$  and equating it to zero, yielding the implicit update equation for the mapping  $p(z|y)$  [CT06]

$$p(z|y) = \frac{p(z) \cdot e^{-\frac{1}{\beta} \cdot \tilde{d}(y, z)}}{\sum_z p(z) \cdot e^{-\frac{1}{\beta} \cdot \tilde{d}(y, z)}}. \quad (3.10)$$

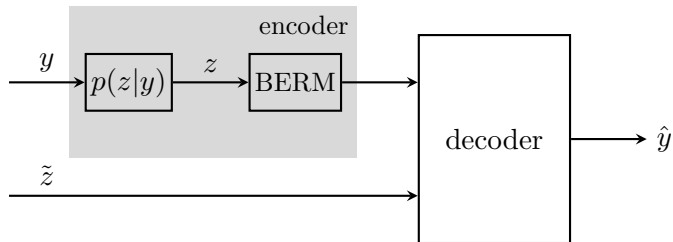


Figure 3.5: Wyner-Ziv system model - source coding with side-information

Due to  $p(z) = \sum_y p(z|y)p(y)$ , equation (3.10) is an implicit equation. In general, this can only be solved by iterative algorithms. One of them is the *Blahut-Arimoto algorithm* [Ari72; Bla72]. The Blahut-Arimoto algorithm splits the original optimization into two alternating minimization steps. In one step,  $p(z|y)$  is optimized while being treated independent of a fixed  $p(z)$ . Respectively, in the next step,  $p(z)$  is optimized while being treated independent of a fixed  $p(z|y)$ .

$$\begin{aligned}
 D(R) &= \min_{p(z)} \min_{p(z|y): I(\mathcal{Y};\mathcal{Z}) \leq R} \mathbb{E}_{\mathcal{Y},\mathcal{Z}} [\tilde{d}(y,z)] \\
 &= \min_{p(z)} \min_{p(z|y): I(\mathcal{Y};\mathcal{Z}) \leq R} \sum_y p(y) \sum_z p(z|y) \tilde{d}(y,z)
 \end{aligned} \tag{3.11}$$

In particular, the alternating steps result in the following procedure: Given a random initialization for  $p(z)$ , calculate  $p(z|y)$  with equation (3.10). Then update the marginal distribution with the updated  $p(z|y)$  by

$$p(z) = \sum_y p(z|y)p(y). \tag{3.12}$$

Repeating this procedure till a certain stopping criterion is fulfilled will solve the original optimization problem. As already mentioned in Subsection 3.1.1, whether this algorithm finds a local or a global optimum depends on the used distortion measure  $\tilde{d}(y,z)$ .

Figure 3.4 illustrates the optimization process of the Blahut-Arimoto algorithm in  $\mathbb{R}^2$  minimizing the Euclidean distance between the two convex sets  $\mathbb{A}$  and  $\mathbb{B}$ . Starting with an arbitrary point  $a_1 \in \mathbb{A}$ , it chooses a point  $b_1 \in \mathbb{B}$  with the smallest distance to  $a_1$ . By fixing this point, the algorithm finds a point  $a_2 \in \mathbb{A}$ , which has the smallest distance to  $b_1$ . This process is repeated and converges to the global optimum since the minimized function is convex [CT84]. This also holds for the case of  $\mathbb{A}$  and  $\mathbb{B}$  being convex sets of probability distributions and the KL divergence being the function to be minimized [Slo02].

## 3.2 Wyner-Ziv Coding

An extension of the general rate-distortion problem where the decoder has access to side-information about the source has first been analyzed in [Wyn78; WZ76]. The Wyner-Ziv (WZ) system model is given in Figure 3.5. The difference to the original rate-distortion problem is that the decoder has additional side-information about  $y$  via the signal  $\tilde{z}$ .

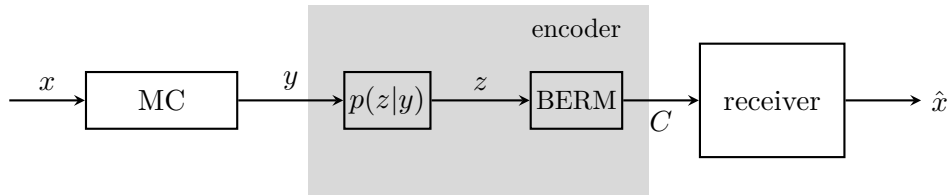


Figure 3.6: General remote sensing setup - encoder compresses noisy signal to forward it over a channel with capacity  $C$  to a receiver

Again, the basic idea is to design a mapping  $p(z|y)$  for the encoder in order to find the lowest rate  $R$ , such that there is an encoder-decoder pair which satisfies  $\mathbb{E}_{\mathcal{Y}, \mathcal{Z}} [\tilde{d}(y, z)] \leq D$ . The optimization problem can be formulated as

$$R_{\text{WZ}}(D) = \min_{p(z|y): \mathbb{E}_{\mathcal{Y}, \mathcal{Z}} [\tilde{d}(y, z)] \leq D} \left[ I(\mathcal{Y}; \mathcal{Z}) - I(\mathcal{Z}; \tilde{\mathcal{Z}}) \right]. \quad (3.13)$$

Due to the Markov chain  $\tilde{\mathcal{Z}} \rightarrow \mathcal{Y} \rightarrow \mathcal{Z}$  this optimization problem can be rewritten as

$$R_{\text{WZ}}(D) = \min_{p(z|y): \mathbb{E}_{\mathcal{Y}, \mathcal{Z}} [\tilde{d}(y, z)] \leq D} I(\mathcal{Y}; \mathcal{Z} | \tilde{\mathcal{Z}}) \quad (3.14)$$

which allows a nice interpretation. Basically, Wyner-Ziv coding tries to minimize the mutual information between  $\mathcal{Y}$  and  $\mathcal{Z}$  given the statistics of  $\tilde{\mathcal{Z}}$ . In other words, it tries to maximize the compression given  $\tilde{\mathcal{Z}}$  while an average distortion shall not exceed a maximum distortion  $D$ . Hence, the compression can exploit that the decoder knows the side-information  $\tilde{\mathcal{Z}}$ , meaning that less information has to be compressed. This Wyner-Ziv coding principle finds application, e.g., in areas like resource-constrained wireless multimedia sensor networks [Xio+10] or in video compression systems [AZG02].

### 3.3 Remote Sensing (Noisy Source Coding)

Remote sensing (RS) or noisy source coding mainly describes the rate-distortion theory for a setup with additional noise, also named as indirect rate-distortion problem [DT62; Sak68; WZ70; Wit80; EG88]. The general remote sensing scenario is depicted in Figure 3.6. It consists of a remote measuring device, i.e., a sensor, trying to observe a physical quantity  $x$  and forwarding it to a distant receiver. The measurement process can be modeled as a memoryless channel (MC) described by the transition probabilities  $p(y|x)$ . As a very simple example, the measurement process could be modeled as AWGN, where the physical quantity  $x$  is corrupted by normally distributed noise  $w$  from various sources. Therefore, the sensor can only observe noisy versions  $y$  of the signal of interest  $x$ , such that  $y = x + w$  with  $w$  denoting the measurement noise. This also motivates the name *noisy source coding*. The sensor has to compress its measurements  $y$  in order to not exceed a maximum compression rate  $R$  to be able to forward them over a link with capacity  $C$ , i.e.,  $R \leq C$ .

Again, the encoder in Figure 3.6 can be realized as a quantizer with a mapping  $p(z|y)$  and a further binary encoding with a rate-matching step. In contrast to the original rate-distortion theory, the distortion  $\tilde{d}(x, z)$  is now measured between  $x$  and  $z$ . However, the compression is still measured between  $y$  and  $z$ . The optimization problem for the rate-distortion function  $R_{RS}(D)$  and the distortion-rate function  $D_{RS}(R)$  to optimize the quantizer mapping  $p(z|y)$  becomes

$$R_{RS}(D) = \min_{p(z|y): \mathbb{E}_{\mathcal{X}, \mathcal{Z}}[\tilde{d}(x, z)] \leq D} I(\mathcal{Y}; \mathcal{Z}) \Leftrightarrow D_{RS}(R) = \min_{p(z|y): I(\mathcal{Y}; \mathcal{Z}) \leq R} \mathbb{E}_{\mathcal{X}, \mathcal{Z}}[\tilde{d}(x, z)]. \quad (3.15)$$

The following derivations will focus on the  $D_{RS}(R)$  function only. Using the method of Lagrangian multipliers, the above equation can be rewritten as

$$L_{RS}[p(z|y)] = \mathbb{E}_{\mathcal{X}, \mathcal{Z}}[\tilde{d}(x, z)] + \beta I(\mathcal{Y}; \mathcal{Z}), \quad (3.16)$$

which has to be minimized. The solution becomes [Ber71]

$$p(z|y) = \frac{p(z) \cdot e^{-\frac{1}{\beta} \cdot \sum_x p(x|y) \cdot \tilde{d}(x, z)}}{\sum_z p(z) \cdot e^{-\frac{1}{\beta} \cdot \sum_x p(x|y) \cdot \tilde{d}(x, z)}}. \quad (3.17)$$

A comparison of the update equation in (3.17) with the one in (3.10) illuminates the similarity of both expressions. The main difference comes from the conditional expectation of the distance measure with respect to the relevant signal, which can be interpreted as the conditional mean

$$d(y, z) \hat{=} \mathbb{E}_{\mathcal{X}|y}[\tilde{d}(x, z)] = \sum_x p(x|y) \cdot \tilde{d}(x, z). \quad (3.18)$$

In order to solve the above equation, the iterative Blahut-Arimoto algorithm [Ari72; Bla72] can be applied using the slightly different average distance measure  $d(y, z)$ . Again, if  $d(y, z)$  is independent of  $p(z|y)$ , the optimization problem is convex, and the Blahut-Arimoto algorithm finds the global optimum. Otherwise, it converges to a local optimum.

### 3.4 The Information Bottleneck Method

The IB method is a clustering framework pairing concepts from machine learning and information theory first introduced by Tishby et al. [TPB99; Slo02; GNT03]. Generally speaking, it is a special case of the remote sensing problem, which also tries to find a trade-off between a compression rate and an average distortion. The connection to the rate-distortion problem is analyzed in [HT07] considering various different distortion measures and their properties. They showed that the IB method is a rate-distortion problem where the Kullback-Leibler divergence is used as a distortion measure. The IB framework finds application in various different fields like pattern recognition, e.g., for

document clustering [ST00], the controversial analysis of neural networks [TZ15; Sax+19], in neuroscience [Sch+01] or in communications, which will be the main focus here. In particular, a rich set of IB applications can be found using mutual-information-based signal processing [Bau+18]. This is a framework that tries to replace conventional arithmetical operations using lookup tables designed by means of the IB method. This can decrease the computational complexity and memory requirements, which makes it very interesting for mobile and low-energy devices. Examples are the analog-to-digital conversion for channels with memory in [Zei10] or the design [Lew+18] and decoding of regular low density parity check (LDPC) codes using IB-optimized lookup tables [LB15; LSB16b; LSB17; LB18; Sta+19; Sta+20a; Sta+20b]. This approach has been extended to irregular LDPC codes in [SLB18b; SLB18a]. Since lookup tables can be quite large and occupy a significant chip area, the authors introduced a strategy to represent these lookup tables in a more efficient way using neural networks in [SLB20]. Other applications deal with channel estimation [Lew+17; KK17a], compressed sensing detectors [FK19] and polar codes [SSB18; SSB19]. In [MSK22], the authors utilize the IB principle to approximate the computation of intermediate quantities within the fast Fourier transform (FFT) for orthogonal frequency division multiplexing (OFDM) systems. In this way, the FFT relies only on quantization indices of IB-optimized lookup tables. The IB framework has been extended to distributed clustering of a common relevant mutual information as well. In [Zei12], cooperative Quantize-and-Forward relaying schemes have been optimized using the IB method. Furthermore, the IB method has been applied in the design of quantizers for compress and forward relay networks [KK17c; KK17b]. An alternating algorithm based on the IB framework was introduced in [CK16a; CK16b] to deal with individual rate constraints in cloud radio access networks. This alternating approach has been adapted for distributed sensor networks in [SK19]. However, it turns out that the solution only focuses on the individual rate constraints of each sensor and does not fulfill the corresponding sum-rates. Furthermore, distributed sensor networks with imperfect forward links have been optimized in [Win14; WMB13; Has+20]. The IB approach has recently been applied in order to define a semantic communication system design in [BBD22]. There, the authors modeled the semantic aspect as a hidden random variable and applied a data-driven approach with neural networks to design the encoder. This encoder shall maximize the mutual information between the semantic random variable and a received signal vector. In the literature, there exist several algorithms to solve the IB optimization problem [TPB99; Slo02; ST99], the iterative IB algorithm, the sequential IB algorithm, an agglomerative IB algorithm, a deterministic annealing-like algorithm, the double maxima algorithm [MA16] or the KL-means algorithm [Kur17], just to mention some examples. The underlying mathematical theory of a deterministic annealing-like algorithm has been intensively studied in [GPD12]. In [Kur17], the author shows that the iterative IB and the KL-means algorithm are algorithmically equivalent for the special case when the focus solely lies on preserving relevant information. In [HWD18c], the equivalence of the double maxima and the KL-means algorithm is proved for the case of perfect forward channels, i.e., no residual error probabilities. A further but different deterministic IB algorithm has been introduced in [SS17]. By introducing a second "trade-off" parame-



ter  $\alpha$ , the authors ensured the optimized mapping to be deterministic. An overview of these algorithms is given in [HWD17] while their asymptotic performance is compared in [Has+17]. In [Che+03], the authors introduced an IB approach for the case where the relevant signal and the observation are jointly multivariate Gaussian distributed. They derived an analytical algorithm to reduce the dimensions of the observations by means of the IB framework resulting in continuous Gaussian distributed compression variables. A parametric version of the iterative IB algorithm has been introduced in [SLB19] requiring the relevant signal to be a Gaussian mixture as well as  $p(y|x)$  to be Gaussian distributed. Note that the compressed variable, in this case, is discrete, compared to [Che+03]. The parametric IB approach can reduce the computational complexity since the KL divergence can be calculated analytically. In [HWD18b], the author utilizes the graph-based affinity propagation algorithm [FD07] to apply an information bottleneck clustering approach and compared corresponding results to the KL-means algorithm. They showed that the affinity propagation algorithm is an efficient way to do the clustering and is competitive to existing approaches. A multivariate extension of the IB approach is given in [Slo02; SFT06]. This approach has been adapted in [HWD19] to develop a novel design framework for the multivariate IB variant in order to find a trade-off between a relevant mutual information and compression sum-rate.

In [Ale+16], the authors introduce a variational approximation of the information bottleneck method, allowing them to parameterize the IB model using a neural network. This variational IB approach has been applied for unsupervised clustering in [UAZ20b]. An overview on the general IB method, including the variational approximation, is given in [ZES20]. In addition, this tutorial paper considers some useful connections to general coding problems, like remote source coding, information combining, or common reconstruction.

### 3.4.1 The Optimization Problem

In principle, the general IB setup is a special case of the remote sensing setup as depicted in Figure 3.6. A sensor tries to measure the relevant signal  $x$ . Its noisy observation  $y$  needs to be compressed, e.g., to forward it over capacity-limited links, while preserving as much information as possible about the original signal  $x$ . In the context of random processes, this problem can also be formulated as follows: Given a random variable  $\mathcal{X}$ , its noisy observation  $\mathcal{Y}$  and the compressed version  $\mathcal{Z}$  forming a Markov chain  $\mathcal{X} \rightarrow \mathcal{Y} \rightarrow \mathcal{Z}$ , the IB method aims to optimize the compression or quantization described by the mapping  $p(z|y)$ . The optimization problem can be defined as minimizing the compression rate  $I(\mathcal{Y}; \mathcal{Z})$  while the relevant mutual information  $I(\mathcal{X}; \mathcal{Z}) \geq \tilde{D}$  does not fall below a certain average distortion  $\tilde{D}$ . Equivalently, it can be described as maximizing the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  such that the compression rate  $I(\mathcal{Y}; \mathcal{Z}) \leq R$  does not exceed a predefined rate  $R$ , e.g., to not exceed a subsequent link capacity  $C$ . This holds since  $p(\hat{x}^n|x)$  and  $p(z^n|x)$  are identical for  $\epsilon$ -typical sequences. The maximization or minimization is done with respect to the mapping  $p(z|y)$ . Naturally,  $|\mathcal{Z}| < |\mathcal{Y}|$  holds

because  $\mathcal{Z}$  is a compressed version of  $\mathcal{Y}$ . Formally, the optimization problem for the  $R_{IB}(\tilde{D})$  and the  $\tilde{D}_{IB}(R)$  can be stated as

$$R_{IB}(\tilde{D}) = \min_{p(z|y): I(\mathcal{X}; \mathcal{Z}) \geq \tilde{D}} I(\mathcal{Y}; \mathcal{Z}) \Leftrightarrow \tilde{D}_{IB}(R) = \max_{p(z|y): I(\mathcal{Y}; \mathcal{Z}) \leq R} I(\mathcal{X}; \mathcal{Z}). \quad (3.19)$$

The following derivations will focus on the  $\tilde{D}_{IB}(R)$  function only. According to [Slo02], the optimization problem can be formulated using the method of Lagrangian multipliers

$$L_{IB}[p(z|y)] = \max_{p(z|y)} I(\mathcal{X}; \mathcal{Z}) - \beta I(\mathcal{Y}; \mathcal{Z}). \quad (3.20)$$

Similar to the general rate-distortion theory, the Lagrange multiplier  $\beta$  in (3.20) can be interpreted as a trade-off parameter between preserving relevant information and compression. In the extreme case where  $\beta = 0$ , the focus solely lies on maximizing the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$ . In [Zei12], the author shows that, in this case, the optimization is a convex maximization problem that has an optimal solution that is deterministic. In the other extreme case for  $\beta \rightarrow \infty$ , the focus lies on compression only. In this case, all elements of  $y$  will be mapped on the same output cluster  $z$ . For the general case, due to  $I(\mathcal{X}; \mathcal{Z}) = H(\mathcal{X}) - H(\mathcal{X}|\mathcal{Z})$  and the fact, that  $H(\mathcal{X})$  does not depend on the mapping  $p(z|y)$ , (3.20) can be reformulated as

$$L_{IB}[p(z|y)] = \min_{p(z|y)} H(\mathcal{X}|\mathcal{Z}) + \beta I(\mathcal{Y}; \mathcal{Z}) \quad (3.21)$$

allowing the direct comparison of remote sensing in (3.16) and the IB method in (3.21). It reveals that the IB method is a special formulation of the remote sensing problem. Since  $H(\mathcal{X}|\mathcal{Z}) = \mathbb{E}_{\mathcal{X}, \mathcal{Z}} [-\log p(x|z)]$  it becomes obvious that the IB method is equivalent to remote sensing using the logarithmic loss function  $\tilde{d}(x, z) = -\log p(x|z)$  as a distortion measure. In this case, distortion minimization means maximization of the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$ . It has to be mentioned that since the distortion  $\tilde{d}(x, z) = -\log p(x|z)$  is a function of the mapping  $p(z|y)$ , the IB optimization problem is a non-convex optimization problem. In particular, it is neither convex nor concave [KY14].

### 3.4.2 Iterative Information Bottleneck Algorithm

The iterative IB algorithm is a standard approach to solve the previously defined IB problem and has been introduced in [Slo02]. The optimization problem in (3.20) can be solved by equating the derivative of the functional  $L_{IB}[p(z|y)]$  with respect to the mapping  $p(z|y)$  to zero. Rearranging the resulting equation in order to have an expression for the desired mapping delivers the implicit update rule

$$p(z|y) = \frac{p(z) \cdot e^{-\frac{1}{\beta} d(y, z)}}{\sum_z p(z) \cdot e^{-\frac{1}{\beta} d(y, z)}} \quad (3.22)$$

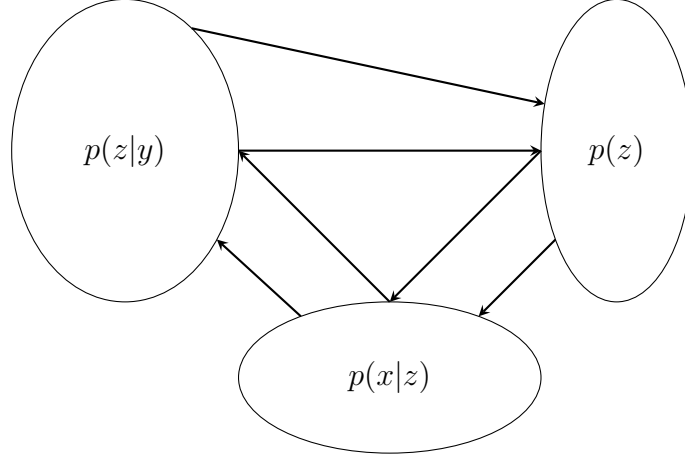


Figure 3.7: Illustration of the IB optimization problem over three convex sets applying the iterative IB algorithm [Slo02]

with  $d(y, z)$  representing a statistical distance obtained by choosing the logarithmic loss distortion measure  $\tilde{d}(x, z) = -\log p(x|z)$

$$\begin{aligned} d(y, z) &= \mathbb{E}_{\mathcal{X}|y} \left[ \log \frac{1}{p(x|z)} \right] \\ &= \mathbb{E}_{\mathcal{X}|y} \left[ \log \frac{p(x|y)}{p(x|z)} \right] + \mathbb{E}_{\mathcal{X}|y} \left[ \log \frac{1}{p(x|y)} \right] \end{aligned} \quad (3.23)$$

$$= D_{\text{KL}} [p(x|y) \| p(x|z)] + \text{const} . \quad (3.24)$$

Since the second term in (3.23) does not depend on the mapping  $p(z|y)$ , it becomes a constant term in (3.24) and can be incorporated into the Lagrange multiplier of the update equation (3.22). Therefore, for the statistical distance  $d(y, z)$ , only the KL divergence remains.

$$d(y, z) := D_{\text{KL}} [p(x|y) \| p(x|z)] \quad (3.25)$$

This KL divergence can be interpreted as the information loss about  $\mathcal{X}$  by representing the relevant information  $\mathcal{X}$  by  $\mathcal{Z}$  instead of  $\mathcal{Y}$ . It has to be emphasised that (3.22) is an implicit solution for the mapping  $p(z|y)$  since  $p(x|z) = \sum_y \frac{1}{p(z)} p(z|y) p(y, x)$  with  $p(z) = \sum_y p(z|y) p(y)$  also depends on the desired mapping. It can be observed that the IB approach is solely based on probability distributions. Therefore,  $z \in \mathcal{Z}$  is just a cluster index, i.e., a categorical variable on a nominal scale, and needs not to be a physical representation of  $x$  or  $y$ . This allows the application of the IB method in non-technical areas where physical representatives are not available or not meaningful, e.g., in text clustering applications.

The implicit equation in (3.22) can be solved by an iterative approach based on the previously described Blahut-Arimoto algorithm in Section 3.1.2. The trick is to sequentially update  $p(z|y)$ ,  $p(z)$ , and  $p(x|z)$  while treating them as independent of each other. If  $\beta > 0$ , the resulting quantizer mapping is stochastic in general, i.e.,  $p(z|y) \in [0, 1]$ . However, for the special case of  $\beta = 0$ , the iterative IB algorithm returns a deterministic mapping,

**Algorithm 1:** Stochastic Iterative IB algorithm [Slo02]

---

```

input      :  $p(x, y), p^{\text{init}}(z|y), \beta > 0, \epsilon$ 
output    :  $p(z|y) \in [0, 1]$ 
1 begin
  initialization:
     $p(z|y)^{(0)} \leftarrow p^{\text{init}}(z|y),$ 
     $l \leftarrow 1$ 
2    $p(x) = \sum_y p(x, y)$ 
3    $p(y) = \sum_x p(x, y)$ 
4    $p(x|y) = p(x, y)/p(y)$ 
5   do
6     // calculate statistical distance  $d(y, z)$  (3.25)
7      $p(z)^{(l)} = \sum_y p(z|y)^{(l-1)} p(y)$ 
8      $p(x|z)^{(l)} = \frac{1}{p(z)^{(l)}} \sum_y p(z|y)^{(l-1)} p(x, y)$ 
9      $D_{\text{KL}}(y, z)^{(l)} = \sum_x p(x|y) \log \frac{p(x|y)}{p(x|z)^{(l)}}$ 
10     $d(y, z)^{(l)} = \beta^{-1} D_{\text{KL}}(y, z)^{(l)} - \log p(z)^{(l)}$ 
11    // update quantizer
12     $p(z|y)^{(l)} = \frac{e^{-d(y, z)^{(l)}}}{\sum_z p(z)^{(l)} e^{-d(y, z)^{(l)}}$ 
13     $l \leftarrow l + 1$ 
14  while  $D_{\text{JS}}^{\Pi}[p^{(l)}(z|y) || p^{(l-1)}(z|y)] > \epsilon$ 

```

---

i.e.,  $p(z|y) \in \{0, 1\}$ . This is very convenient from an implementation point of view since a deterministic mapping  $p(z|y)$  can be implemented by simple lookup tables. It has to be mentioned that the implementation of the iterative IB algorithm assumes  $\mathcal{Y}$  to be a discrete random variable. Hence, it generally requires a previous discretization of  $y$  with an appropriately large resolution.

Figure 3.7 illustrates the IB optimization over three convex sets using the iterative IB algorithm. In each step, two convex sets are kept fixed while optimizing the third one, such that the overall IB functional of (3.20) is minimized.

For the stochastic case, when  $\beta > 0$ , the complete iterative IB algorithm is given in Algorithm 1. For a specific  $\beta$ , an input distribution  $p(x, y)$  and an initial mapping  $p^{\text{init}}(z|y)$ , the quantizer mapping  $p(z|y)$  is iteratively updated until a specific convergence criterion is fulfilled. First, the statistical distance measure (3.25) is calculated, which is then used to update the quantizer mapping. The JS divergence with weights of  $\Pi = \{0.5, 0.5\}$  between successive quantizer mappings can be used as a convergence criterion. If this is smaller than or equal to a predefined accuracy  $\epsilon$ , the algorithm stops and returns the optimized quantizer mapping  $p(z|y)$ .

For the deterministic case, when  $\beta = 0$ , the sum in the denominator of the update equation (3.22) is dominated by the smallest KL divergence. Hence, for each sample  $y$ , the mapping

**Algorithm 2:** Deterministic Iterative IB algorithm

---

```

input      :  $p(x, y), p^{\text{init}}(z|y), \epsilon$ 
output    :  $p(z|y) \in \{0, 1\}$ 
1 begin
  initialization:
     $p(z|y)^{(0)} \leftarrow p^{\text{init}}(z|y),$ 
     $l \leftarrow 1$ 
2    $p(x) = \sum_y p(x, y)$ 
3    $p(y) = \sum_x p(x, y)$ 
4    $p(x|y) = p(x, y)/p(y)$ 
5   do
6     // calculate KL Divergence  $D_{\text{KL}}(y, z)$  (3.25)
7      $p(z)^{(l)} = \sum_y p(z|y)^{(l-1)} p(y)$ 
8      $p(x|z)^{(l)} = \frac{1}{p(z)^{(l)}} \sum_y p(z|y)^{(l-1)} p(x, y)$ 
9      $D_{\text{KL}}(y, z)^{(l)} = \sum_x p(x|y) \log \frac{p(x|y)}{p(x|z)^{(l)}}$ 
10    // find minimum of  $D_{\text{KL}}(y, z)$  for all samples  $y_i \in \mathbb{Y}$ 
11    for  $y_i \in \mathbb{Y}$  do
12       $z^*(y_i)^{(l)} = \arg \min_z D_{\text{KL}}(y, z)^{(l)}$ 
13      // update quantizer for specific  $y_i$ 
14       $p(z^*(y_i)^{(l)}|y_i)^{(l)} = 1$ 
15     $l \leftarrow l + 1$ 
16  while  $D_{\text{JS}}[p^{(l)}(z|y) || p^{(l-1)}(z|y)] > \epsilon$ 

```

---

$p(z|y)$  tends to zero for all  $z$  except for the one with the smallest KL divergence. Only this value tends to one, leading to

$$z^*(y) = \arg \min_z D_{\text{KL}} [p(x|y) || p(x|z)] , \quad (3.26)$$

and the update equation

$$p(z|y) = \begin{cases} 1 & \text{for } z(y) = z^*(y) \\ 0 & \text{else.} \end{cases} \quad (3.27)$$

Since in the deterministic case, the focus solely lies on preserving relevant information and  $\beta = 0$  has a fixed value, the compression is defined by choosing an appropriate cardinality  $|\mathbb{Z}|$ . The deterministic iterative IB algorithm is given in Algorithm 2. The difference to the stochastic case lies in the update procedure of the mapping  $p(z|y)$ , which only requires the minimum search.

The IB framework is often analyzed using the relevance-compression plane. Here, the compression rate  $I(\mathcal{Y}; \mathcal{Z})$  is depicted versus the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  showing how much distortion is introduced for a specific compression. Figure 3.8 depicts this relevance-compression plane after optimizing the quantizer mapping  $p(z|y)$  using the iterative IB algorithm. For different cardinalities  $|\mathbb{Z}|$  and a 4-ASK relevant signal with

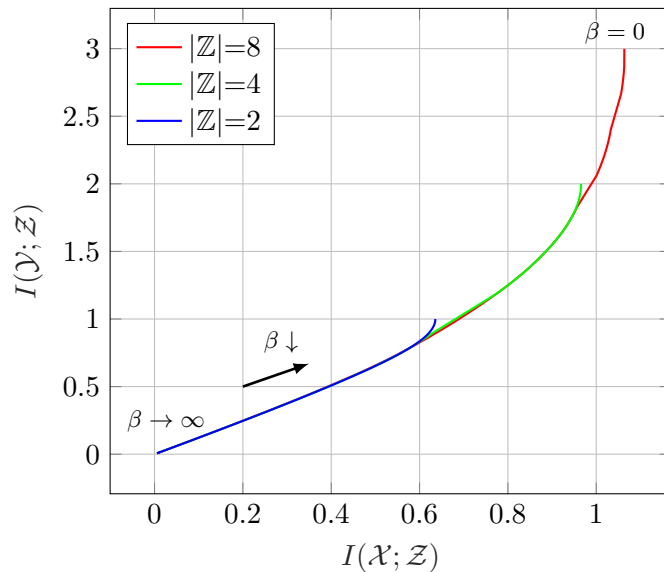


Figure 3.8: Relevance-Compression plane using the iterative IB algorithm for different cardinalities  $|\mathbb{Z}|$ ,  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}| = 256$ , SNR  $\gamma = 6$  dB

$|\mathbb{X}| = 4$ , the Lagrange multiplier has been adapted in order to obtain different relevance-compression curves. By decreasing  $\beta$ , the trade-off focuses more on the preservation of relevant information, leading to higher relevant mutual information and higher compression rates, i.e., less compression. The most upper point of each curve is obtained by choosing  $\beta = 0$ . The curves saturate on different relevant mutual information due to the limited cardinality  $|\mathbb{Z}|$ . Naturally, lower cardinalities saturate earlier.

Figure 3.9 shows a comparison of the stochastic and deterministic iterative IB algorithm. The solid curve represents the stochastic case, achieved by using the iterative IB algorithm for varying values of  $\beta$  with a fixed output cardinality of  $|\mathbb{Z}| = 16$ . The square marks represent the results achieved with the deterministic iterative IB algorithm using  $\beta = 0$ . The different points result from varying the number of output clusters  $|\mathbb{Z}|$  from 1 to 8. Both algorithms are performed for ASK relevant signals with different cardinalities and different SNRs  $\gamma$ . In general, the performance of the stochastic iterative IB is slightly superior to the deterministic iterative IB. This confirms the statement of [Slo02] that stochastic mappings usually perform better since the global optimum is stochastic in general. However, the difference becomes smaller when the optimization point lies in the area where the curve saturates. Finally, it can be observed that the deterministic iterative IB algorithm produces approximately equally distributed output clusters, at least for lower SNRs or higher relevant signal cardinalities. This can be confirmed by looking at the compression rates  $I(\mathcal{Y}; \mathcal{Z}) = H(\mathcal{Z})$ . It has to be mentioned that both algorithms only obtain local optima, and therefore each solution depends on the initial mapping.

**Measurement SNR Mismatch:** The last simulation revealed that the IB method maximizes the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  while still maintaining a desired compression rate  $I(\mathcal{Y}; \mathcal{Z})$ . However, the question arises of how robust this approach is against measurement SNR mismatches, as the signal-to-noise ratio  $\gamma$  might not be known in ad-

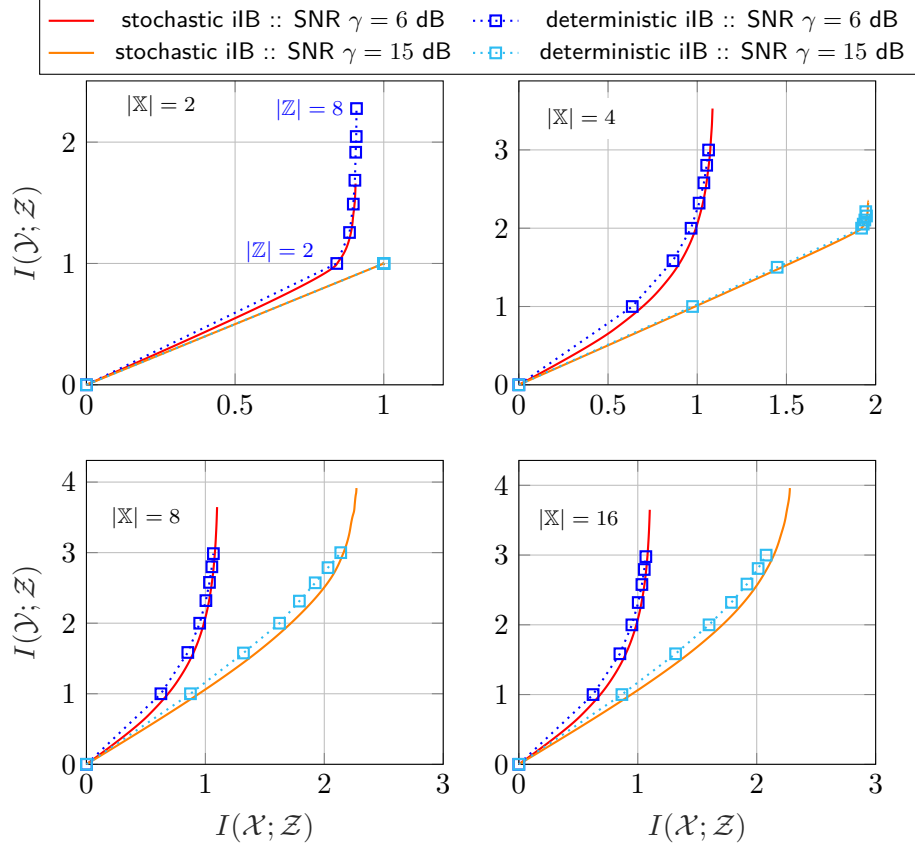


Figure 3.9: Relevance-Compression plane comparing stochastic iterative IB and the deterministic iterative IB algorithm for different  $|\mathbb{X}|$ ,  $|\mathbb{Y}| = 512$  and  $\text{SNR } \gamma = \{6, 15\}$  dB

vance or does change over time. Therefore, this paragraph considers the case where the optimization SNR does not match the actual application SNR. To be more specific, given a fixed application measurement signal-to-noise ratio  $\gamma_{\text{app}}$  with a corresponding channel likelihood  $p(y|x)$ , a quantizer  $\tilde{p}(z|y)$  optimized for a different SNR  $\gamma_{\text{opt}}$  with  $\tilde{p}(y|x)$  is applied. The resulting relevant mutual information is given as

$$\tilde{I}(\mathcal{X}; \mathcal{Z}) = \mathbb{E}_{\mathcal{X}, \mathcal{Z}} \left[ \log_2 \frac{\tilde{p}(z|x)}{\tilde{p}(z)} \right] \quad (3.28)$$

while the resulting compression rate is given as

$$\tilde{I}(\mathcal{Y}; \mathcal{Z}) = \mathbb{E}_{\mathcal{Y}, \mathcal{Z}} \left[ \log_2 \frac{\tilde{p}(z|y)}{\tilde{p}(z)} \right]. \quad (3.29)$$

Note that the required pmfs can be obtained by

$$\tilde{p}(z, x) = \sum_y \tilde{p}(z|y)p(y|x)p(x) \quad (3.30)$$

and

$$\tilde{p}(z) = \sum_x \tilde{p}(z, x). \quad (3.31)$$

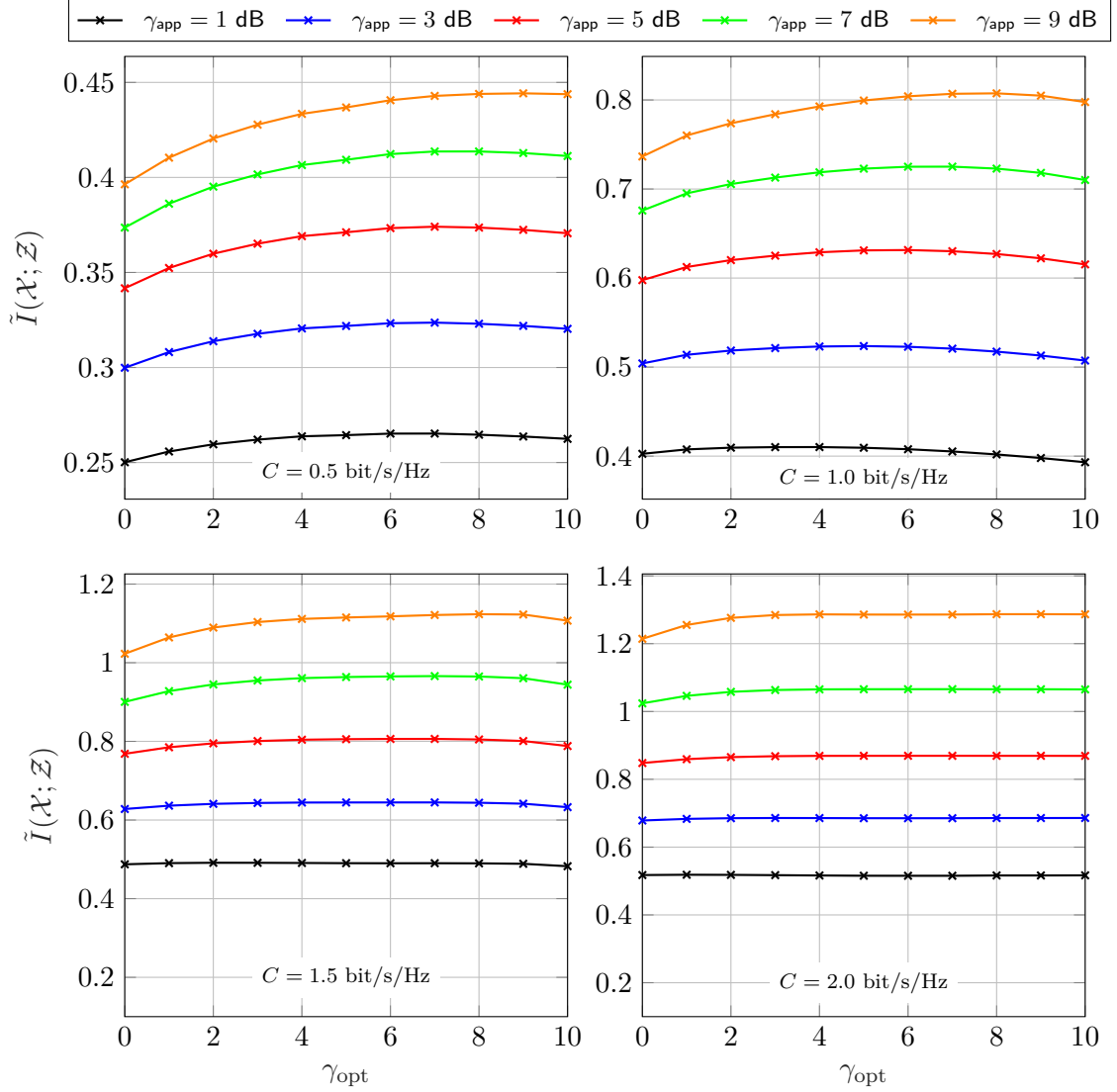


Figure 3.10: Relevant mutual information in the case of measurement SNR mismatch using the IB method with  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}| = 512$  and  $|\mathbb{Z}| = 4$

Figure 3.10 illustrates four scenarios, each for a different rate constraint  $C$ . These rate constraints are chosen to match different levels of compression in order to forward the information of a 4-ASK relevant signal to the receiver. Hence,  $C = 0.5$  bit/s/Hz represents a strict rate constraint requiring a strong compression, while  $C = 2.0$  bit/s/Hz is a loose rate constraint. The overall relevant mutual information  $\tilde{I}(\mathcal{X}; \mathcal{Z})$  is depicted versus the optimization SNR  $\gamma_{\text{opt}}$  for different application SNRs  $\gamma_{\text{app}}$ . Naturally, in all plots, the overall relevant mutual information  $\tilde{I}(\mathcal{X}; \mathcal{Z})$  is larger for a larger application SNR  $\gamma_{\text{app}}$ . The general expectation is that the maximum of each curve occurs at  $\gamma_{\text{opt}} = \gamma_{\text{app}}$ . In the case of a rate constraint of  $C = 2.0$  bit/s/Hz, it can be observed that a measurement channel mismatch has not a big impact. Here, the overall relevant mutual information has a large plateau where the mismatch does not decrease the performance. Therefore, for larger target rates or even no rate constraints, the measurement channel mismatch is not a problem. For  $C = 1.5$  bit/s/Hz, there still exists a large plateau in which the mismatch has



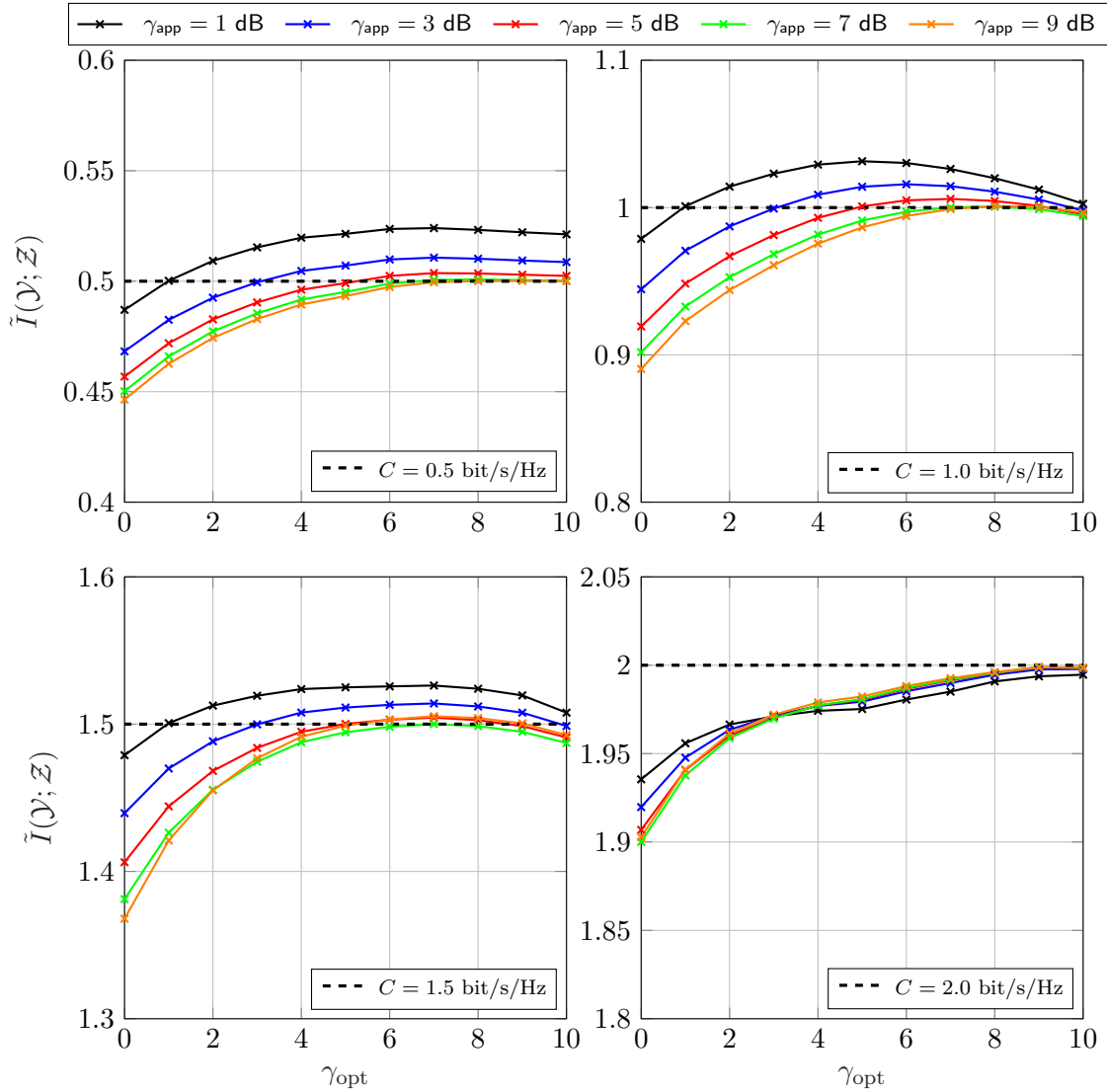


Figure 3.11: Compression rate in the case of measurement SNR mismatch using the IB method with  $|\mathcal{X}| = 4$ ,  $|\mathcal{Y}| = 512$  and  $|\mathcal{Z}| = 4$

no influence. However, the influence of a measurement channel mismatch slightly increases, especially for larger application SNRs. Here, it can be observed that the maximum lies on the optimization point, while a mismatch can lead to a slight performance loss in relevant mutual information  $\tilde{I}(\mathcal{X}; \mathcal{Z})$ . This effect increases for stricter rate constraints. For  $C = 1.0$  bit/s/Hz, the maximum still lies very close to the optimization point. However, the plateau observed for larger rate constraints is no longer apparent. Here, the measurement channel mismatch has a much higher impact on the overall performance. If the channel capacity is very low, i.e.,  $C = 0.5$  bit/s/Hz, it can be observed that for  $\gamma_{\text{app}} = \{1, 3, 5\}$  dB, the maxima no longer match the optimization point.

To understand this behavior, the compression rates have to be analyzed as well. The measurement channel mismatch can also lead to exceeding the required compression rate. Therefore, Figure 3.11 illustrates the compression rate  $\tilde{I}(\mathcal{Y}; \mathcal{Z})$  for the same setups as in Figure 3.10. The black dashed lines represent the capacity bound  $C$ . Starting with

$C = 2.0$  bit/s/Hz, it becomes obvious that, in this case, no real problem occurs regarding the compression rates  $\tilde{I}(\mathcal{Y}; \mathcal{Z})$ . Here, the capacity  $C = 2.0$  bit/s/Hz is fulfilled for all simulated application SNRs. This confirms the observations from Figure 3.10 that measurement SNR mismatch has a low impact on larger channel capacities. For all other depicted capacities  $C = \{0.5, 1.0, 1.5\}$  bit/s/Hz, the measurement SNR mismatch significantly affects the compression rates. The case where  $\tilde{I}(\mathcal{Y}; \mathcal{Z})$  falls below the capacity  $C$  is uncritical. However, this does not hold if the resulting compression rate is larger than the capacity  $\tilde{I}(\mathcal{Y}; \mathcal{Z}) > C$  since the rate constraint is violated. This effect seems to occur mainly for low application SNRs  $\gamma_{\text{app}}$  when the optimization has been performed for a larger optimization SNR  $\gamma_{\text{opt}}$ . If the optimization SNR  $\gamma^{\text{opt}}$  is better than the application SNR  $\gamma^{\text{app}}$  the true channel likelihood  $p(y|x)$  is broader than  $\tilde{p}(y|x)$  leading to an increase in compression rate  $\tilde{I}(\mathcal{Y}; \mathcal{Z})$ . This overshoot of  $\tilde{I}(\mathcal{Y}; \mathcal{Z})$  over the capacity  $C$  might be the reason why the maximum in Figure 3.10 is not exactly on the optimization point  $\gamma_{\text{opt}}$ . In conclusion, it can be stated that for loose rate constraints, the measurement SNR mismatch has not a big impact. However, for stricter rate constraints, when the measurements have to be stronger compressed, the impact increases, leading to possible losses when applying a quantizer for a wrong SNR. Moreover, measurement SNR mismatch can result in exceeding the channel capacity  $C$ , making reliable communication impossible. Therefore, for lower target capacities  $C$ , it becomes more important to match the optimization SNR. Hence, if the application SNR is unknown during the optimization or changes significantly over time, the above simulations suggest two possible strategies to deal with possible mismatches. First, a compression rate back-off could be considered during the optimization. In this case, the optimization has to be performed for a lower rate constraint than the actual capacity  $C$ , such that the overshoot does not violate this rate constraint. However, it might be hard to define the required back-off in advance. A second approach is to optimize for a low SNR. In this case, it is very likely that even if the measurement channel has a significant mismatch, the compression rate is still lower than the rate constraint  $C$ . It has to be mentioned that both approaches result in a loss of relevant mutual information.

### 3.5 Channel-Aware Information Bottleneck Approach

The original IB approach assumes error-free transmission over a rate-limited forward channel using a capacity-achieving coding scheme. In practical approaches, there always remain residual errors due to finite-length codes. Consequently, the quantizers designed with the previously described algorithms might no longer be optimal in practical systems. In [Win14; WMB13] the author proposed the Channel Aware Information Bottleneck (CAIB) algorithm, i.e., an IB based algorithm, which takes into account the residual error probabilities modeled as a DMC. This setup can be considered as a special case of the remote sensing setup and can be modeled as depicted in Figure 3.12. The CAIB algorithm has been investigated for vector compression in [Has+20] and compared to other IB algorithms in [HWD17; Has+17]. The graph-based affinity propagation algorithm already applied for the original IB setup in [HWD18b] has been extended for the channel-aware

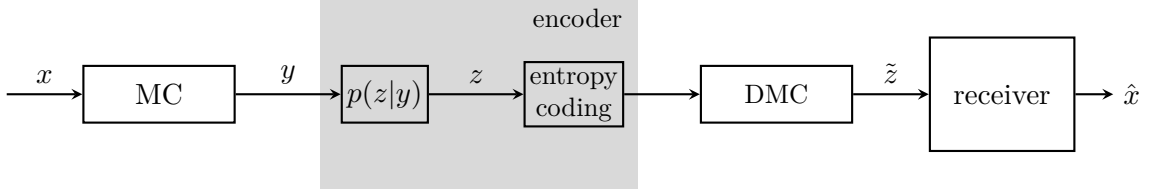


Figure 3.12: Schematic illustration of the CAIB setup modelling residual error probabilities as a DMC

case in [HWD18a]. In [MWD19], the authors applied this CAIB for a single user C-RAN system.

As mentioned above, the residual error probabilities can be modeled as a DMC with a specific transition probability  $p(\tilde{z}|z)$ . Within this thesis, it is assumed that the number of inputs and outputs of the DMC is identical, i.e.,  $|\mathcal{Z}| = |\tilde{\mathcal{Z}}|$ . Since we do not assume an error-free transmission, the impact of the channel has to be taken into account. Hence, the relevant mutual information can be defined as  $I(\mathcal{X}; \tilde{\mathcal{Z}})$ . With a given statistic of the DMC, the compression rate is specified implicitly. Therefore, the focus solely lies on maximizing the relevant mutual information leading to the optimization problem

$$p(z|y)^* = \arg \max_{p(z|y)} I(\mathcal{X}; \tilde{\mathcal{Z}}) . \quad (3.32)$$

In [Win14], the author showed that since (3.32) is a convex maximization problem with respect to the mapping  $p(z|y)$ , the optimal mapping is deterministic, i.e.,  $p(z|y)^* \in \{0, 1\}$ . Following the derivations in [Win14], with  $I(\mathcal{X}; \tilde{\mathcal{Z}}) = I(\mathcal{X}; \mathcal{Y}) - I(\mathcal{X}; \mathcal{Y}|\tilde{\mathcal{Z}})$  and the fact that  $I(\mathcal{X}; \mathcal{Y})$  is independent of the mapping  $p(z|y)$ , the optimization problem in (3.32) can be reformulated to

$$p(z|y)^* = \arg \min_{p(z|y)} I(\mathcal{X}; \mathcal{Y}|\tilde{\mathcal{Z}}) . \quad (3.33)$$

Note, that  $I(\mathcal{X}; \mathcal{Y}|\tilde{\mathcal{Z}})$  is minimized if  $\tilde{\mathcal{Z}}$  contains information about  $\mathcal{Y}$  and therefore also about  $\mathcal{X}$ . This confirms that (3.32) and (3.33) are equivalent regarding the optimization of  $p(z|y)$ . Using the definition of the KL divergence, the mutual information in (3.33) can be decomposed to

$$\begin{aligned} I(\mathcal{X}; \mathcal{Y}|\tilde{\mathcal{Z}}) &= \sum_y p(y) \sum_z p(z|y) \underbrace{\sum_{\tilde{z}} p(\tilde{z}|z) D_{\text{KL}} [p(x|y) \| p(x|\tilde{z})]}_{C(y,z)} \\ &= \sum_y p(y) \sum_z p(z|y) C(y, z) \end{aligned} \quad (3.34)$$

**Algorithm 3:** Channel-Aware IB algorithm [Win14]

---

```

input      :  $p(x, y), p(\tilde{z}|z), p^{\text{init}}(z|y), \beta, \epsilon$ 
output    :  $p(z|y) \in \{0, 1\}$ 
1 begin
  initialization:
     $p(z|y)^{(0)} \leftarrow p^{\text{init}}(z|y),$ 
     $l \leftarrow 1$ 
2  $p(x|y) = p(x, y) / \sum_x p(y)$ 
3 do
4   // calculate KL Divergence  $D_{\text{KL}}(y, z)$ 
5    $p(x, y, \tilde{z})^{(l)} = \sum_z p(\tilde{z}|z) p(z|y)^{(l-1)} p(x, y)$ 
6    $p(x|\tilde{z})^{(l)} = \sum_y p(x, y, \tilde{z})^{(l)} / \sum_{y,x} p(x, y, \tilde{z})^{(l)}$ 
7    $D_{\text{KL}}(y, z)^{(l)} = \sum_x p(x|y) \log \frac{p(x|y)}{p(x|\tilde{z})^{(l)}}$ 
8   // calculate  $C(y, z)$  and find cluster
9    $C(y, z)^{(l)} = \sum_{\tilde{z}} p(\tilde{z}|z) D_{\text{KL}}(y, z)^{(l)}$ 
10  for  $y_i \in \mathbb{Y}$  do
11     $z^*(y_i)^{(l)} = \arg \min_z C(y_i, z)^{(l)}$ 
12    // update quantizer for specific  $y_i$ 
13     $p(z^*(y_i)^{(l)}|y_i)^{(l)} = 1$ 
14   $l \leftarrow l + 1$ 
15 while  $D_{\text{JS}}^{\text{II}}[p^{(l)}(z|y) || p^{(l-1)}(z|y)] > \epsilon$ 

```

---

Due to the non-negativity of the KL divergence, the mutual information  $I(\mathcal{X}; \tilde{\mathcal{Z}})$  is minimized by minimizing the function  $C(y, z)$  for each value  $y \in \mathbb{Y}$ . Thus, the optimal cluster  $z^*$  for a specific  $y$  can be determined by

$$z^*(y) = \arg \min_z C(y, z) \quad \forall y \in \mathbb{Y}. \quad (3.35)$$

A pseudo-code of the CAIB algorithm is given in Algorithm 3.

Figure 3.13 illustrates the performance of the CAIB algorithm. It depicts the relevant mutual information  $I(\mathcal{X}; \tilde{\mathcal{Z}})$  for different measurement signal-to-noise ratios  $\gamma$ . The DMC is modeled as a symmetric matrix  $|\mathbb{Z}| \times |\tilde{\mathbb{Z}}|$  with  $|\mathbb{Z}| = |\tilde{\mathbb{Z}}|$ , defined by a reliability parameter  $\epsilon$ . For each symbol, the probability of a correct transmission is given as  $1 - \epsilon$ , whereas all other transitions have the probability  $\frac{\epsilon}{N-1}$ . The relevant signal is chosen to be a 4-ASK signal, while other cardinalities are chosen to  $|\mathbb{Y}| = 64$  and  $|\mathbb{Z}| = 8$ . As an initial mapping, a maximum output entropy quantizer is applied.

The gray-shaded area represents the non-achievable region, since  $I(\mathcal{X}; \mathcal{Y}) \geq I(\mathcal{X}; \tilde{\mathcal{Z}})$  due to the data processing inequality. The dashed-dotted line represents the performance for a perfect transmission channel. Here, the original IB with  $\beta = 0$  has been performed without residual error probabilities modeled by a DMC. The solid and dashed lines show the performance of the CAIB and the original IB for a DMC with a specific  $\epsilon$ . It becomes obvious that a forward channel with a higher average error probability results in a lower overall performance, independent of the measurement SNR. However, the performance of

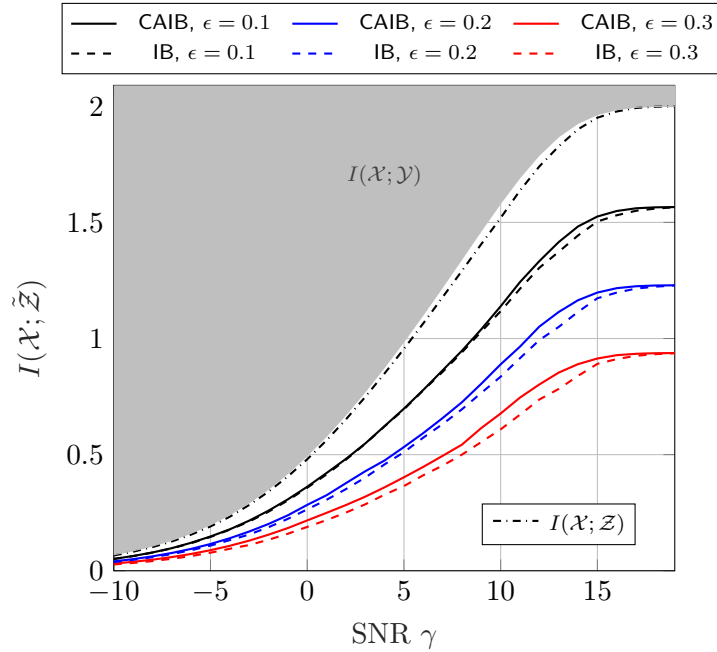


Figure 3.13: Relevant mutual information  $I(\mathcal{X}; \tilde{\mathcal{Z}})$  for different measurement signal-to-noise ratios and an artificially modeled DMC comparing the Channel-Aware Information Bottleneck to the original IB approach;  $|\mathcal{X}| = 4$ ,  $|\mathcal{Y}| = 64$ ,  $|\mathcal{Z}| = 8$

the CAIB is superior to the performance of the original IB algorithm, especially for DMCs with larger  $\epsilon$  corresponding to larger average error probabilities.

Figure 3.14 illustrates the result for the same simulation but with a different model for the DMC. The  $|\mathcal{Z}| = 8$  output clusters are directly mapped on 8-ASK symbols such that neighboring cluster correspond to neighboring symbols. The 8-ASK symbols are assumed to be equally probable, resulting in decision boundaries for a hard decision centered in the middle between adjacent symbols. The forward channel assumes to be an AWGN channel with a specific SNR  $\gamma_{\text{FC}}$ . This SNR is chosen such that it results in the same DMC capacity as for the previous simulation with the DMC modeled by the reliability parameter  $\epsilon$ . It can be observed that the gap between the CAIB approach and the original IB approach is slightly smaller than for the previous setup. In addition, the overall loss introduced by the DMC is smaller than in Figure 3.13. This can be explained by the different error probabilities given in the DMC matrix  $p(\tilde{z}|z)$ . In Figure 3.14, neighboring symbols are mixed up more likely, while others are not. This causes a minor information loss compared to the artificial DMC of Figure 3.13.

**Measurement Channel Mismatch:** Similar to the original IB case, the question arises of how robust the CAIB algorithm performs when measurement SNR assumptions do not hold. Therefore, the same analysis as in Figure 3.10 is performed, but for the CAIB algorithm. In particular, given a fixed application measurement SNR  $\gamma_{\text{app}}$  with  $p(y|x)$ , a

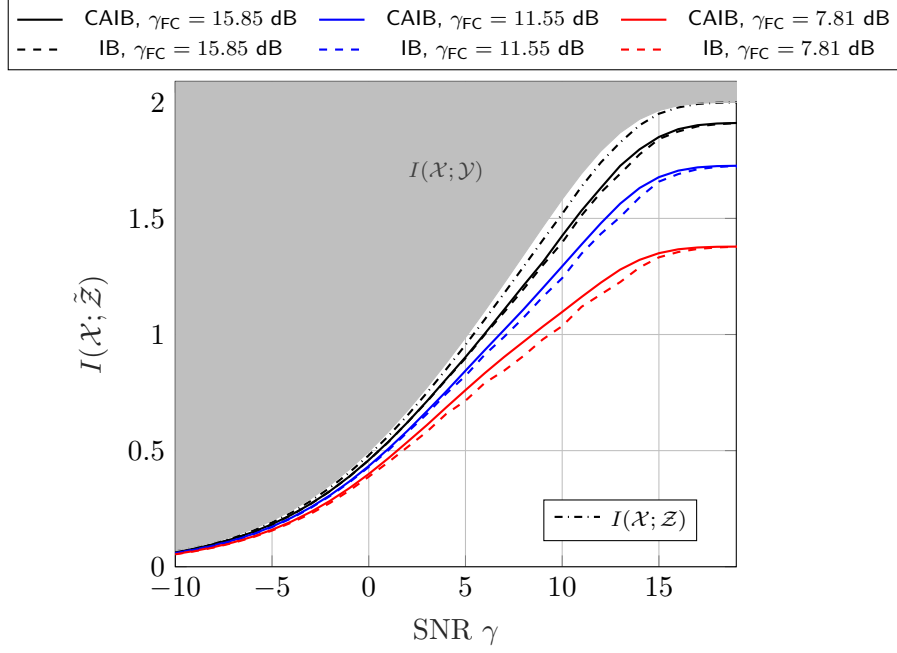


Figure 3.14: Relevant mutual information  $I(\mathcal{X}; \tilde{\mathcal{Z}})$  for different measurement signal-to-noise ratios and an AWGN modeled DMC comparing the Channel-Aware Information Bottleneck to the original IB approach;  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}| = 64$ ,  $|\mathbb{Z}| = 8$

quantizer  $\tilde{p}(z|y)$  optimized for a different SNR  $\gamma_{\text{opt}}$  with  $\tilde{p}(y|x)$  is applied. In this case, the resulting relevant mutual information can be determined by

$$\tilde{I}(\mathcal{X}; \tilde{\mathcal{Z}}) = \mathbb{E}_{\mathcal{X}, \tilde{\mathcal{Z}}} \left[ \log_2 \frac{\tilde{p}(\tilde{z}|x)}{\tilde{p}(\tilde{z})} \right] \quad (3.36)$$

with

$$\tilde{p}(\tilde{z}, x) = \sum_y \sum_z p(\tilde{z}|z) \tilde{p}(z|y) p(y|x) p(x) \quad (3.37)$$

and

$$\tilde{p}(\tilde{z}) = \sum_x \tilde{p}(\tilde{z}, x) . \quad (3.38)$$

Figure 3.15 illustrates the relevant mutual information  $\tilde{I}(\mathcal{X}; \tilde{\mathcal{Z}})$  versus the optimization SNR  $\gamma_{\text{opt}}$  for different application SNRs and different forward channels. Similar to Figure 3.14, the forward channel is described by a specific SNR  $\gamma_{\text{FC}}$ . Note that  $\gamma_{\text{FC}} \rightarrow \infty$  corresponds to a perfect forward channel with  $\bar{p}_e = 0$ , i.e., it does not introduce any errors. In this case, the main difference to Figure 3.10 is that the CAIB algorithm performs a deterministic mapping, and no rate constraint defined by a capacity  $C$  has to be matched. Hence, it performs a deterministic iterative IB algorithm with  $\beta = 0$ . It can be observed that for  $\gamma_{\text{FC}} \rightarrow \infty$ , the CAIB algorithm is quite robust against measurement channel mismatches. For most application SNRs, the relevant mutual information is nearly constant over a wide range. When introducing an erroneous forward channel with

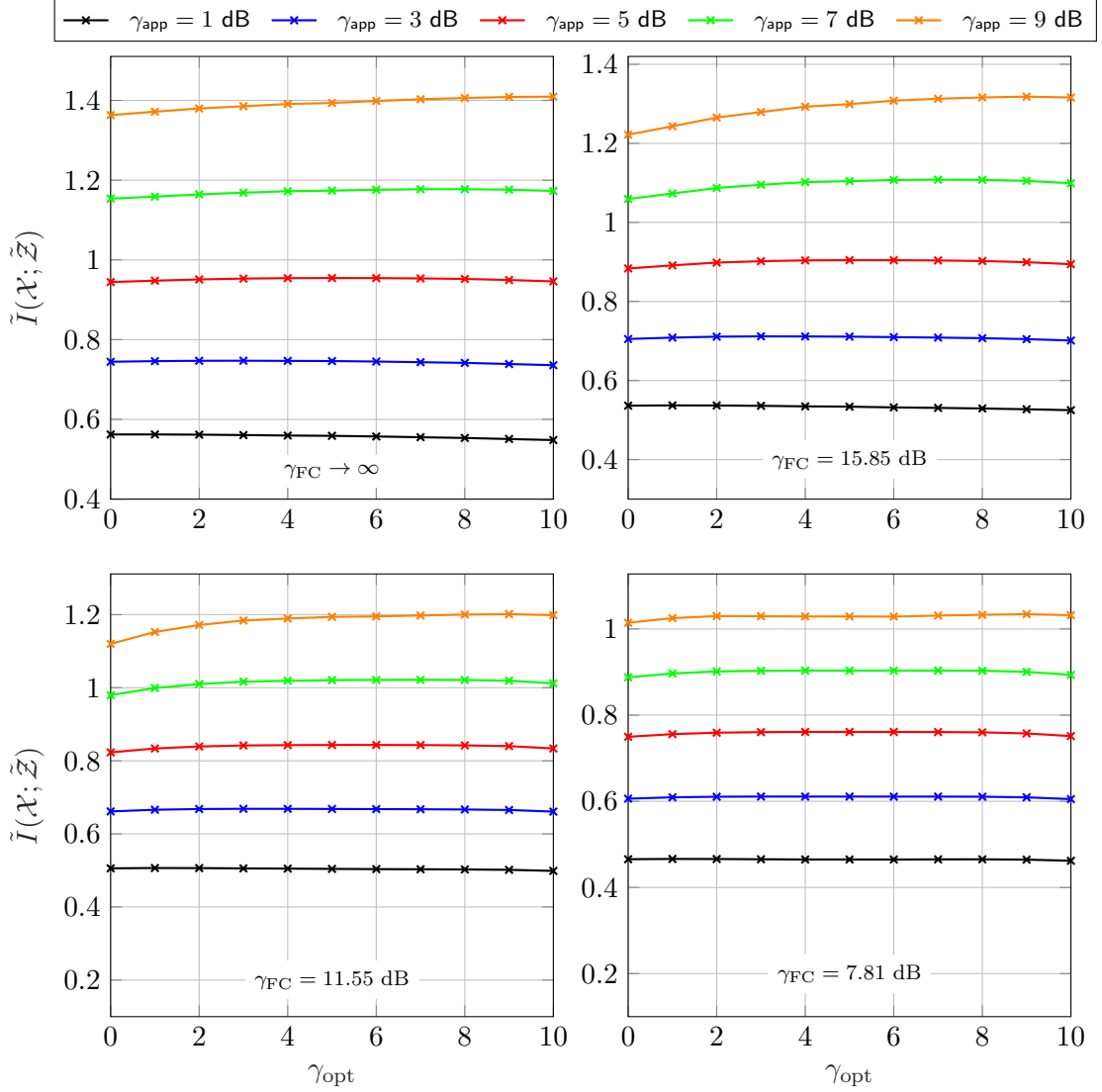


Figure 3.15: Measurement channel mismatch applying the CAIB algorithm with  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}| = 512$ ,  $|\mathbb{Z}| = 8$

$\gamma_{\text{FC}} = \{15.85, 11.55, 7.81\}$  dB, it can be observed that the relevant mutual information  $\tilde{I}(\mathcal{X}; \tilde{\mathcal{Z}})$  is still constant over a wide range. However, for  $\gamma_{\text{FC}} = 15.85$  dB and an application SNR of  $\gamma_{\text{app}} = 9$  dB, the loss introduced by a low optimization SNR is increased compared to the error-free case. By increasing the average error probability, i.e., reducing the capacity of the DMC, in the two lower plots, this effect seems to be reduced. However, this might be caused by the lower overall relevant information. Hence, the loss is not as pronounced as for higher overall relevant information. In conclusion, it can be stated that the CAIB is quite robust against measurement channel mismatch. Only for the case of  $\gamma_{\text{opt}} \ll \gamma_{\text{app}}$  a loss is introduced. This can be explained by the narrow application pmf  $p(y|x)$  being represented by only a few clusters since the optimization has been performed for a much broader pmf.

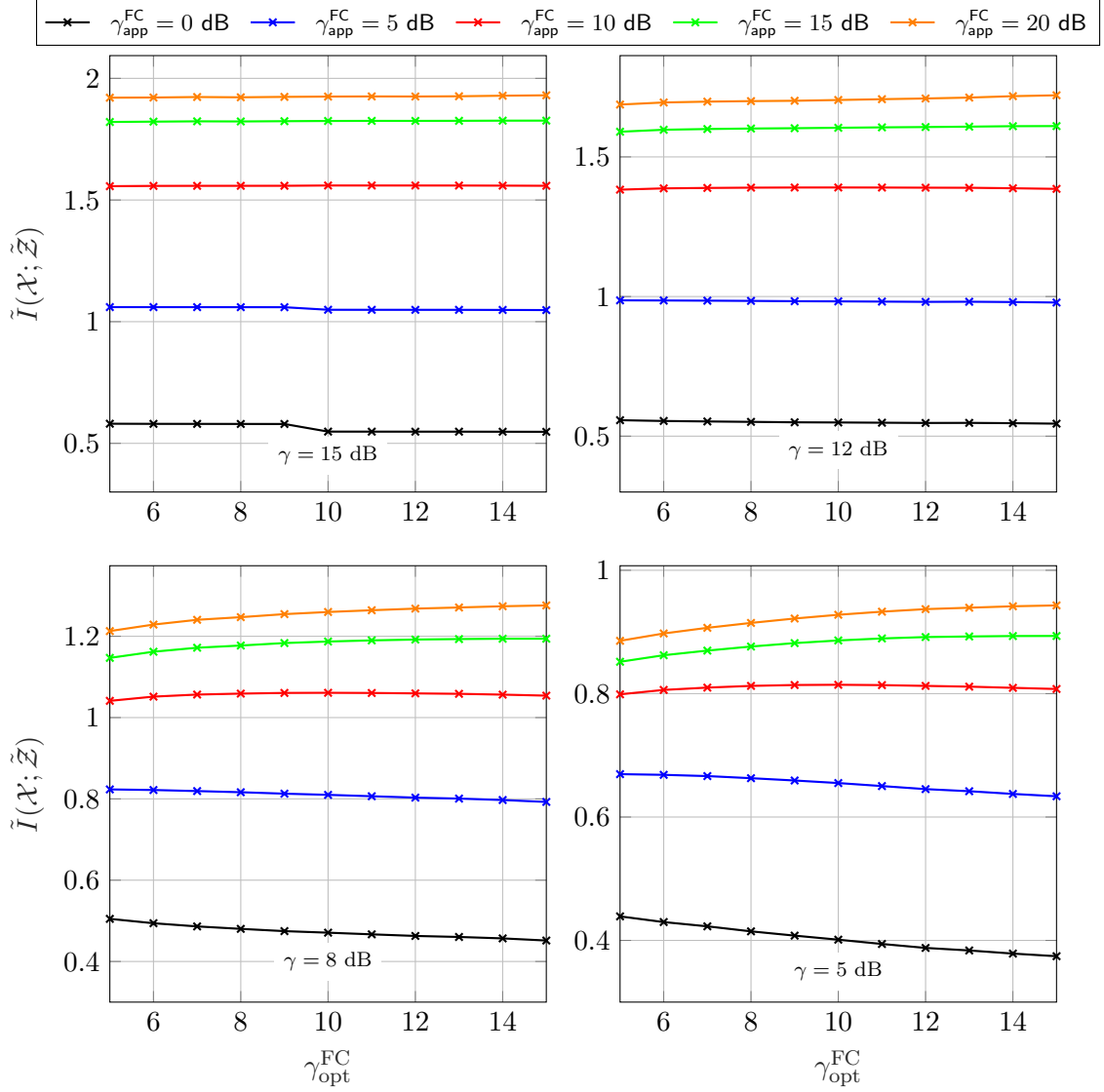


Figure 3.16: Forward channel mismatch applying the CAIB algorithm with  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}| = 512$ ,  $|\mathbb{Z}| = 8$

**Forward Channel Mismatch:** In the case of the CAIB algorithm, not only the measurement mismatch is of interest, but also a mismatch of the forward channel  $p(\tilde{z}|z)$ . Similar to Figure 3.14, the forward channel is defined by a specific SNR  $\gamma^{\text{FC}}$ . A quantizer  $\tilde{p}(z|y)$  optimized for a specific optimization SNR  $\gamma_{\text{opt}}^{\text{FC}}$  defining  $\tilde{p}(\tilde{z}|z)$  is applied for a different application SNR  $\gamma_{\text{app}}^{\text{FC}}$  defining  $p(\tilde{z}|z)$ . The relevant mutual information can be determined as before.

Figure 3.16 illustrates the relevant mutual information  $\tilde{I}(\mathcal{X}; \tilde{\mathcal{Z}})$  versus the optimization SNR  $\gamma_{\text{opt}}^{\text{FC}}$  for different application SNRs  $\gamma_{\text{app}}^{\text{FC}}$ . The different plots correspond to different measurement SNRs  $\gamma = \{15, 12, 8, 5\}$  dB. It can be observed that for a large measurement SNR, the forward channel mismatch has a very low influence. Here, the relevant mutual information is nearly constant over a wide range of  $\gamma_{\text{opt}}^{\text{FC}}$ . When reducing the measurement SNR, the influence of the mismatch increases. It can be observed that the maximum is



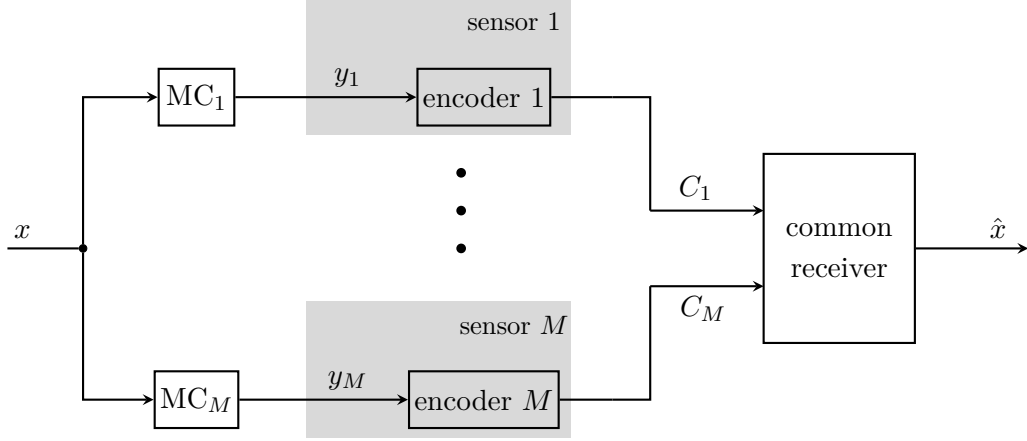


Figure 3.17: System model for the non-cooperative remote sensing problem

always located on the point where the optimization SNR matches the application SNR

$$\gamma_{\text{opt}}^{\text{FC}} = \gamma_{\text{app}}^{\text{FC}}.$$

## 3.6 Distributed Remote Sensing

### 3.6.1 Non-Cooperative Remote Sensing - The CEO Problem

**System Model:** The scalar remote sensing problem can be extended to the distributed case, as depicted in Figure 3.17. Here,  $M$  distributed sensors measure noisy versions of the same signal  $x$  generated by the process of interest  $\mathcal{X}$ , in the following referred to as the *relevant signal*. It has to be mentioned that the communication among sensors is not possible in this system model defining the non-cooperative remote sensing. Since the sensors are independent of each other, the sensing process can be described by  $M$  statistically independent MCs. A very simple example of this abstract sensing process is that zero-mean additive white Gaussian measurement noise disturbs the measurements  $y_m = x + w_m$ . In this case  $p(y_m|x)$  would be defined by the measurement SNR given by  $\gamma_m = \frac{\sigma_x^2}{\sigma_{w_m}^2}$ , with  $\sigma_x^2$ ,  $\sigma_{w_m}^2$  denoting signal and noise variances, respectively. Equivalent to the scalar case, each sensor has to compress its measurements  $y_m$  in order to be able to forward them without any further loss over individual capacity-limited links to a common receiver. These links to the common receiver are referred to as forward links. The encoding process contains a scalar quantization defined by the mapping  $p(z_m|y_m)$  and a subsequent BERM step, depending on  $p(z_m|y_m) \in \{0, 1\}$  being deterministic or  $p(z_m|y_m) \in [0, 1]$  being stochastic. Therefore, the actual index, which is transmitted to the common receiver, is a compressed version of  $z_m$ . The system model, given in Figure 3.17 with a specific encoder from Figure 3.18, forms the Markov property illustrated in Figure 3.19. It becomes obvious, that given  $\mathcal{X}$  all variables in  $\mathcal{Z}$  and  $\mathcal{Y}$  are conditionally independent such that

$$p(\mathbf{z}, \mathbf{y}, x) = p(x) \prod_{m=1}^M p(z_m|y_m)p(y_m|x) \quad (3.39)$$

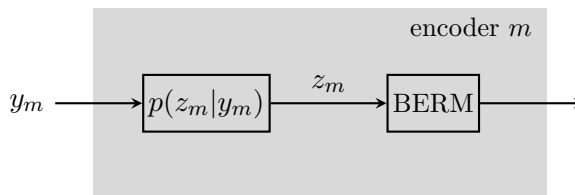


Figure 3.18: Encoding process the non-cooperative remote sensing problem

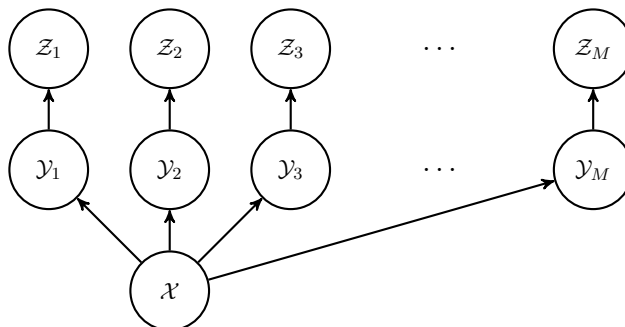


Figure 3.19: Illustration of the Markov property of the CEO system model

holds. Finding the rate region for this non-cooperative distributed sensing scenario is generally known as the CEO problem.

**The CEO Problem:** The CEO problem was originally proposed by T. Berger et al. in [BZV96]. Here, different so-called agents observe the same process of interest. Note that each agent just has a limited view on this process, which is why their observations are only independent subjective interpretations of it. The agents aim to forward their observations to a central entity, the CEO. However, the CEO has limited resources and cannot process all observations of each agent, which is why the agents have to compress their observations in order to be as informative as possible about the process of interest. The CEO itself tries to reconstruct the process of interest out of the compressed observations of all agents.

There exists a wide range of investigations concerning the CEO problem in the literature of the last decades. It has been considered for different relevant signals and different distortion measures. For the general case, the CEO rate region is still unknown. The original paper of T. Berger et al. [BZV96] investigated the error rate performance for a discrete source with the Hamming distance as a distortion measure and exhibited an inevitable loss due to non-cooperating sensors. In [EG18], for arbitrary distortion measures, a scaling law on the sum-rate distortion function has been derived. Thus, in a network with  $M$  sensors, the distortion scales with  $1/M$  for analog transmission but only with  $1/\log(M)$  for digital transmission. Moreover, many investigations exist for the quadratic Gaussian CEO problem, considering jointly Gaussian signals and the MSE distortion measure [Ooh98; VB97; Jun+04; PTR04; Ooh05; WTV08]. More precisely, in [Ooh98] Y. Oohama et al. analytically derived an asymptotic version of the sum-rate distortion function when the number of agents, also called encoders, goes to infinity. In [VB97], the authors investigated the influence of cooperating and non-cooperating encoders on the distortion measure. It turns out that the MSE distortion asymptotically decreases with the reciprocal sum-rate

$R$  for non-cooperating encoders. In contrast, it decays exponentially, i.e., with  $2^{-2R}$  for cooperating encoders. The non-asymptotic case was first investigated in [Jun+04]. The authors derived an upper bound on the sum-rate distortion function and showed that it is tight when each encoder has the same measurement SNR. In [PTR04], [Ooh05] and [WTV08] the authors characterized the complete rate region for the quadratic Gaussian CEO problem.

A multivariate version of the Gaussian CEO problem has been investigated amongst others in [UAZ18; UAZ20a; WC12; CW11; XW16]. In particular, the vector Gaussian CEO problem using a logarithmic loss distortion measure has been analyzed in [UAZ18; UAZ20a], where the authors provide a complete characterization of the rate region for their scenario. In [CW11] and [WC12] the authors derived an outer bound of the rate region for the general vector Gaussian CEO problem and the general L-terminal vector Gaussian CEO problem, respectively. Moreover, they showed that this outer bound matches with the Berger-Tung inner bound in specific regions. In [XW16], the authors provide a complete characterization of the rate region of the vector Gaussian CEO problem using a trace distortion measure.

In addition, many investigations exist for the CEO problem with an arbitrary discrete source distribution and the logarithmic loss distortion measure. In [CW14], Courtade and Weissman completely characterize the rate region for this setup. However, they only provide a theoretical definition of this rate region. More practical algorithmic solutions have been proposed in [EZ18; UAZ17]. The authors introduced a Blahut-Arimoto-like algorithm, which sequentially optimizes the mappings of the distributed devices. However, in [EZ18], they focused on the sum-rate distortion measure only, which does not allow individual rate adjustments. In [UAZ17], the authors proposed an algorithmic approach to finding the complete rate region for the two encoder case. Here, they reduce the complexity by splitting the original rate region into two simpler regions. These simpler regions can be efficiently calculated and used to reconstruct the original one. The approach of [EZ18] has been extended in [HWD20] to optimize a single user equipment (UE) C-RAN system and deal with individual rate adjustments of the fronthaul links. A different algorithmic approach has been proposed in [CK16a; CK16b]. Here, the authors considered a C-RAN system with multiple UEs and developed an alternating approach to optimize the quantizers of each radio access point (RAP) incorporating individual fronthaul capacities. Therefore, they introduced an alternating information bottleneck algorithm combined with an alternating bi-section search. A data-driven approach to solving the CEO problem based on machine learning with neural networks is introduced in [AZ21; ZA20]. Here, the authors consider the case of the relevant distribution to be unknown for optimization. Finally, an overview of existing IB variants and applications, as well as the distributed CEO scenario, is given in [ZES20].

In general, finding analytical solutions is often not possible since the numerical computation of the rate region quickly becomes challenging due to the exponentially growing complexity in the number of sensors. Therefore, most results restrict to just a few sensors for their simulations or the sum-rate constraint only. Moreover, many of these algorithmic solutions determine the rate region for a given distortion. In practical systems, however, it

is often of interest to minimize the distortion for given rate constraints. Although these approaches are equivalent considering the theoretical characterization of the rate-distortion region, they differ in their algorithmic design. This thesis concentrates on the latter case.

**Theoretical Bounds and Rate-Distortion Region:** The main focus in the literature concerning the CEO problem is to find the complete rate region. However, this region is generally hard to determine, as it is only defined for particular distortion measures and relevant signal distributions. The definition of the rate region generally tries to find the rates for a specific target distortion. As stated above, the theoretical characterization of the rate-distortion region is equivalent if one is looking for the minimum distortion given certain rate tuples or the reverse. The whole characterization results in a set of simultaneously achievable tuples of rates and distortion. As this thesis considers the CEO problem with a logarithmic loss distortion measure, it is worthwhile to have a short look at the theoretical definition of the rate-distortion region of [CW14].

The rate-distortion region for the CEO problem  $\mathbb{R}_{\text{CEO}}^*$  is defined as the set of all rate-distortion tuples  $(R_1, \dots, R_M, D)$  being achievable. In order to define this region, they introduce an inner bound for the CEO rate-distortion region, which turns out to be a special case of the *Berger-Tung Inner Bound* [Ber71], first proposed by T. Berger and S. Y. Tung, respectively. The inner bound for the rate-distortion region with  $M$  sensors is given as

$$I(\mathcal{Y}_{\mathbb{S}}; \mathcal{Z}_{\mathbb{S}} | \mathcal{Z}_{\bar{\mathbb{S}}}, \mathcal{Q}) \leq \sum_{m \in \mathbb{S}} C_m \quad \forall \quad \mathbb{S} \subseteq \{1, 2, \dots, M\} \quad (3.40)$$

$$H(\mathcal{X} | \mathcal{Z}, \mathcal{Q}) \leq D. \quad (3.41)$$

It is defined for any distribution  $p(x, \mathbf{y}, \mathbf{z}, \mathbf{q}) = p(\mathbf{q})p(x) \cdot \prod_{m=1}^M p(z_m | y_m, \mathbf{q}) \cdot p(y_m | x)$ . Equation (3.40) defines the rate constraints with  $\mathbb{S}$  being a subset of all sensors and  $\bar{\mathbb{S}}$  containing all sensors, not in  $\mathbb{S}$ . These rate constraints must hold for any subset  $\mathbb{S} \subseteq \{1, 2, \dots, M\}$ . It becomes obvious that the number of rate constraints increases exponentially with the number of sensors. The parameter  $C_m$  represents the individual rate constraint of a specific sensor  $m$ . Equation (3.41) defines the distortion constraint. The parameter  $D$  represents the maximum permitted distortion introduced by compression. In both equations (3.40) and (3.41), the parameter  $\mathcal{Q}$  is a time-sharing parameter defining the time allocated for different coding strategies. In this way, a rate region can be obtained by convex combination of all extreme points resulting from specific coding strategies. This inner bound defines a subset of the complete rate-distortion region  $\mathbb{R}_{\text{CEO}}^i \subseteq \mathbb{R}_{\text{CEO}}^*$ . Therefore, all rate-distortion tuples defined by this inner bound are achievable, i.e., they provide feasible solutions. However, it has to be emphasized that the inner bound does not provide the complete rate-distortion region.

Based on this inner bound, T. A. Courtade and T. Weissman proposed a corresponding outer bound  $\mathbb{R}_{\text{CEO}}^{\circ}$  for the CEO rate region [CW14], which is defined as

$$\left[ \sum_{m \in \mathbb{S}} I(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{X}, \mathcal{Q}) + H(\mathcal{X} | \mathcal{Z}_{\mathbb{S}}, \mathcal{Q}) - D \right]^+ \leq \sum_{m \in \mathbb{S}} C_m \quad \forall \mathbb{S} \subseteq \{1, 2, \dots, M\} \quad (3.42)$$

$$H(\mathcal{X} | \mathcal{Z}, \mathcal{Q}) \leq D \quad (3.43)$$

with  $[\cdot]^+ = \max(0, \cdot)$ . This outer bound is again defined for any distribution  $p(x, \mathbf{y}, \mathbf{z}, \mathbf{q}) = p(\mathbf{q})p(x) \cdot \prod_{m=1}^M p(z_m | y_m, \mathbf{q}) \cdot p(y_m | x)$ . By definition, this outer bound is a superset of the original rate-distortion region  $\mathbb{R}_{\text{CEO}}^{\circ} \supseteq \mathbb{R}_{\text{CEO}}^*$ . Therefore, not all points of the outer bound lie inside the rate-distortion region  $\mathbb{R}_{\text{CEO}}^*$ , i.e., not all points represent feasible solutions. However, the authors showed that each extreme point of the outer bound is *dominated* by a point in the inner bound. That means that for each extreme point of the outer bound with a rate-distortion tuple  $(R_1, \dots, R_M, D) \in \mathbb{R}_{\text{CEO}}^{\circ}$ , there exists a point in the inner bound with a corresponding rate-distortion tuple  $(\tilde{R}_1, \dots, \tilde{R}_M, \tilde{D}) \in \mathbb{R}_{\text{CEO}}^{\text{i}}$  for which  $\tilde{R}_m \leq R_m \forall m$  and  $\tilde{D} \leq D$  holds. More precisely, let  $\mathcal{P}_D$  be the polytope defined by the rate constraints of the outer bound in (3.42). They showed that each extreme point of  $\mathcal{P}_D$  is dominated by a point in the inner bound  $\mathbb{R}_{\text{CEO}}^{\text{i}}$  that has at most the distortion  $D$ . In other words, for a specific extreme point with a rate-distortion tuple  $(R_1, \dots, R_M, D) \in \mathbb{R}_{\text{CEO}}^{\circ}$  there exists a point in the inner bound  $\mathbb{R}_{\text{CEO}}^{\text{i}}$ , which achieves the same distortion with lower rates. Thus, they conclude that the outer bound is a subset of the inner bound  $\mathbb{R}_{\text{CEO}}^{\circ} \subseteq \mathbb{R}_{\text{CEO}}^{\text{i}}$ . Finally, if additionally  $\mathbb{R}_{\text{CEO}}^{\text{i}} \subseteq \mathbb{R}_{\text{CEO}}^*$  and  $\mathbb{R}_{\text{CEO}}^{\circ} \supseteq \mathbb{R}_{\text{CEO}}^*$  holds, the inner and the outer bound must match and define the complete rate-distortion region. However, it has to be emphasized that this is a theoretical description of the CEO rate-distortion region for the given setup. Although the authors in [CW14] showed that each extreme point of the outer bound is dominated by a point in the inner bound, it is not obvious how to obtain this specific point. In other words, an algorithm based on the outer bound will still find solutions that might not lie inside the rate region, although there might exist a dominating point in the inner bound.

**Suboptimal Independent Optimization Approach:** Considering the non-cooperative distributed sensing scenario in Figure 3.17 with an encoder given in Figure 3.18, the easiest way to optimize the quantizer of each sensor is to use a scalar IB method previously defined in Section 3.4. In this way, one optimization problem has to be solved for each sensor

$$\begin{aligned} L_{\text{IB}}^{(1)} &= I(\mathcal{X}; \mathcal{Z}_1) - \beta_1 I(\mathcal{Y}_1; \mathcal{Z}_1) \\ &\vdots \\ L_{\text{IB}}^{(M)} &= I(\mathcal{X}; \mathcal{Z}_M) - \beta_M I(\mathcal{Y}_M; \mathcal{Z}_M). \end{aligned}$$

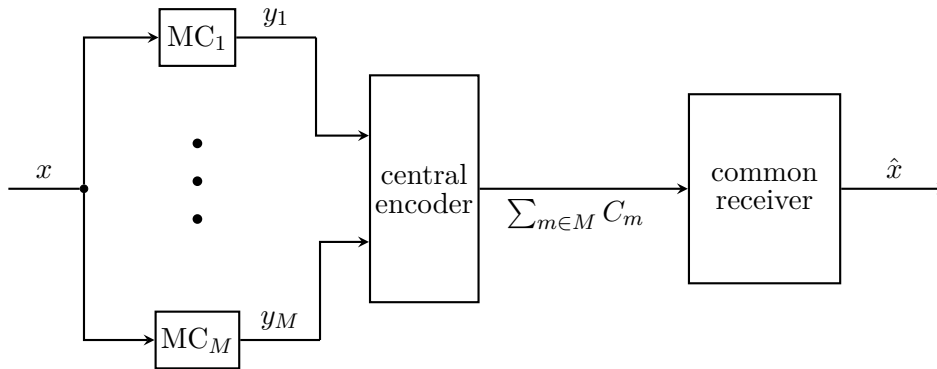


Figure 3.20: System model for the fully cooperative remote sensing scenario modeled by a centralized compression approach

For the  $m$ -th sensor, the parameter  $\beta_m$  has to be adjusted in order to fulfill the given rate constraint  $I(\mathcal{Y}_m; \mathcal{Z}_m) < C_m$ . The optimization approach for sensor  $m$  can be solved by taking the derivative w.r.t. the mapping  $p(z_m|y_m)$  and equating it to zero. This results in the update equation of the scalar IB method given in Section 3.4. Applying a Blahut-Arimoto algorithm leads to local optimal solutions for each independent optimization. However, the independently designed scalar quantizers do not exploit the fact that each sensor measures the same relevant signal, i.e., no Wyner-Ziv coding is applied. Therefore, this approach only serves as a lower bound for the performance of the proposed algorithms in the next chapters.

### 3.6.2 Fully Cooperative Remote Sensing

**System Model:** The fully cooperative Chief Executive Officer (fcCEO) scenario is an extension of the original CEO scenario. Each sensor observes its own measurement  $y_m$  of the same relevant signal  $x$ . This uncompressed measurement can be forwarded directly to all other sensors via perfect inter-sensor links. The exchange of measurements is performed in a distinct cooperation phase resulting in all sensors having access to all measurements. After this phase, the sensors can locally encode the measurements, i.e., they generate a common compression index fulfilling the sum-rate constraint  $C_{\text{sum}} = \sum_{m=1}^M C_m$ . Therefore, it is assumed that each sensor has access to all forward channel conditions of all other sensors. The compression index  $z$  is split into  $M$  parts, taking into consideration the link capacities  $C_m$ . In a forwarding phase, each sensor transmits its corresponding part to the common receiver. Hence, the encoding process contains two distinct phases, a cooperation phase in which sensors exchange uncompressed measurements and a forwarding phase in which all sensors forward a part of the compressed observations in a coordinated way to the common receiver.

**A Centralized Information Bottleneck Approach:** The fcCEO scenario can be modeled as depicted in Figure 3.20. Here, a single device in a central position performs the compression. Therefore, it has access to all observations  $\mathbf{y} = [y_1 \dots y_M]^T$ . The central encoder illustrated in Figure 3.21 includes a quantization with the mapping  $p(z|\mathbf{y})$  and

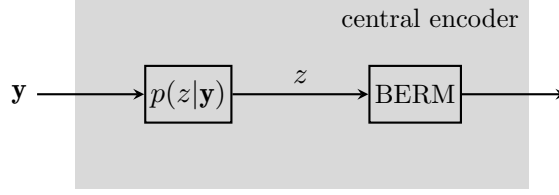


Figure 3.21: Encoder for the fully cooperative remote sensing problem

a binary encoding step with rate-matching, depending on whether  $p(z|\mathbf{y})$  is deterministic or stochastic. By choosing the number of output clusters as  $|\mathcal{Z}| = \prod_{m=1}^M |\mathcal{Z}_m|$  and the single link capacity from the central quantizer to the receiver as  $C_{\text{sum}} = \sum_{m=1}^M C_m$  this centralized model in Figure 3.20 is equivalent to the case of having  $M$  distributed sensors. However, it allows a much easier optimization approach, as only a single quantizer has to be optimized. Applying the IB principle, the central quantizer can be designed in order to compress the vector  $\mathbf{y}$  onto a cluster index  $z$  using the mapping  $p(z|\mathbf{y})$ , which motivates the name Centralized Information Bottleneck (CIB) for the algorithmic approach to find solutions in a fcCEO scenario. The optimization problem can be formulated as

$$L_{CIB}[p(z|\mathbf{y})] = \max_{p(z|\mathbf{y})} I(\mathcal{X}; \mathcal{Z}) - \beta I(\mathcal{Y}; \mathcal{Z}). \quad (3.44)$$

Taking the derivative w.r.t. the mapping  $p(z|\mathbf{y})$  and equating it to zero results in the update equation

$$p(z|\mathbf{y}) = \frac{p(z) \cdot e^{-\frac{1}{\beta} \cdot d(\mathbf{y}, z)}}{\sum_z p(z) \cdot e^{-\frac{1}{\beta} \cdot d(\mathbf{y}, z)}} \quad (3.45)$$

with  $d(\mathbf{y}, z)$  representing the KL divergence

$$d(\mathbf{y}, z) = D_{\text{KL}}[p(x|\mathbf{y}) \| p(x|z)]. \quad (3.46)$$

Applying a Blahut-Arimoto algorithm, this approach can be solved, resulting in local optimal solutions.

In the case where the measurement processes of Figure 3.20 are modeled as additive noise, the centralized quantizer can be reduced to a simple scalar quantizer with the mapping  $p(z|\bar{y})$ . Maximum ratio combining of all inputs  $y_m$  delivers a scalar sufficient statistics

$$\bar{y} = \sum_{m=1}^M \gamma_m \cdot y_m$$

of the desired relevant signal  $x$  with an overall SNR  $\gamma = \sum_m \gamma_m$ . The scalar quantizer can be designed by a standard IB algorithm. Note that reducing (3.44) to a scalar optimization problem is very convenient from an implementation point of view.

### 3.7 Discussion

The basic concepts of lossy data compression have been introduced in this chapter. The original work of Shannon in [Sha48; Sha59] finds a trade-off between an average distortion and a compression. It has been extended for lossy compression with side-information applying the Wyner-Ziv coding principle [Wyn78; WZ76]. The original rate-distortion problem has been extended to noisy sources being the basis for the IB principle, which finds a trade-off between a relevant information and a compression rate. Several connections between the IB problem and the noisy source coding problem have been discussed. The iterative IB algorithm was given as an algorithmic approach to solve this problem. This algorithm is based on the Blahut-Arimoto algorithm defined in [Bla72; Ari72]. Moreover, a special case of the iterative IB algorithm has been introduced, resulting in deterministic mappings. The performance of the IB approach has been discussed, considering the relevant compression curve for stochastic and deterministic mappings. Moreover, the measurement SNR mismatch for different forward channel capacities has been analyzed. It turned out that for larger capacities of the forward channel, the measurement SNR mismatch has not a big impact. Decreasing the capacity increases this impact leading to possible losses when operating a quantizer on a wrong SNR. In addition, this mismatch can result in exceeding the forward channel capacity. The general IB principle has been extended to the case of erroneous forward channels [Win14]. It has been shown that incorporating the forward channel in the optimization problem can increase the overall performance of the system. Moreover, the CAIB algorithm is quite robust to measurement and forward channel mismatches. The scalar remote sensing scenario has been extended to the distributed case introducing the non-cooperative CEO scenario. The suboptimal independent optimization approach simply optimizes each sensor by applying the scalar IB principle. Here, no Wyner-Ziv coding is applied. Since all sensors observe noisy versions of the same relevant signal, this approach is highly suboptimal. Finally, the fully cooperative remote sensing scenario has been introduced. Moreover, to find the solution for this scenario, a centralized approach containing a single compression device in a central position has been considered. After receiving all measurements in a cooperation phase, it forwards the compressed measurements in the forwarding phase to the common receiver. This centralized approach is beneficial since maximum ratio combining can be applied to reduce the fully cooperative scenario to a scalar IB optimization problem, significantly reducing computational complexity.



## Chapter 4

# Algorithmic Solutions for Non-Cooperative Distributed Remote Sensing

Subsection 3.6.1 introduced the general CEO problem as well as a suboptimal independent optimization approach based on the IB method. For the same non-cooperative CEO scenario, this chapter introduces an algorithmic solution exploiting Wyner-Ziv coding, which outperforms the independent IB optimization. Derivations and results of this chapter are published in [Ste+21a; Ste+21b]. Inspired by the work of Courtade and Weismann [CW14], Section 4.1 introduces a greedy optimization approach assuming a specific optimization order. Consequently, this approach allows the formulation of one optimization problem for each sensor exploiting the statistics of previously optimized sensors by Wyner-Ziv coding. In Section 4.2, the Greedy Distributed Information Bottleneck (GDIB) algorithm is introduced as an algorithmic approach for an offline optimization of each sensor, allowing individual rate adjustments for each forward link. The performance and robustness of this algorithm are investigated for different scenarios. Since the optimization process of the GDIB algorithm depends on the network size, it suffers from the curse of dimensionality in larger networks. In order to allow the optimization even for larger networks, a reduced-memory complexity GDIB algorithm is introduced in Section 4.3 using a sequential compression scheme. Naturally, it is analyzed, in which case this compression leads to performance degradation. Finally, Section 4.4 introduces an algorithmic approach that adapts the original GDIB algorithm to deal with imperfect forward channels.

### 4.1 A Greedy Distributed Information Bottleneck Approach

This section introduces a greedy optimization approach proposed in [Ste+21b; SK21], which can be applied to find feasible solutions of the non-cooperative CEO scenario. The optimization approach is based on the inner bound of the CEO rate region, defined in (3.40) and (3.41). Therefore, the solutions do not represent the complete rate region of the CEO problem but represent regions that may be strictly smaller.

The distortion constraint of the inner bound of the CEO rate region in (3.41) can be reformulated using the relevant mutual information  $I(\mathcal{X}; \mathcal{Z}|\mathcal{Q})$ , since the expectation of the logarithmic loss function is  $H(\mathcal{X}|\mathcal{Z}, \mathcal{Q}) = H(\mathcal{X}|\mathcal{Q}) - I(\mathcal{X}; \mathcal{Z}|\mathcal{Q})$  and  $H(\mathcal{X}|\mathcal{Q})$  does not depend on the mapping  $p(z_m|y_m, \mathbf{q})$ . In this way, the optimization problem resembles the IB optimization problem and can be formulated as

$$\begin{aligned} & \max_{\mathbb{P}} I(\mathcal{X}; \mathcal{Z}|\mathcal{Q}) \text{ s.t. } I(\mathcal{Y}_{\mathbb{S}}; \mathcal{Z}_{\mathbb{S}}|\mathcal{Z}_{\bar{\mathbb{S}}}, \mathcal{Q}) \leq \sum_{m \in \mathbb{S}} C_m \\ & \forall \mathbb{S} \subseteq \{1, 2, \dots, M\} \quad , \quad \bar{\mathbb{S}} = \{1, 2, \dots, M\} \setminus \mathbb{S}. \end{aligned} \quad (4.1)$$

Here, the goal is to maximize the relevant mutual information  $I(\mathcal{X}; \mathcal{Z}|\mathcal{Q})$  while simultaneously fulfilling the rate constraints defined in (3.40). In equation (4.1), the set  $\mathbb{P} = \{p(z_1|y_1, \mathbf{q}), \dots, p(z_M|y_M, \mathbf{q})\}$  represents the set of all quantizer mappings. In order to simplify the algorithmic approach to find the extreme points of the solution space of (4.1), it can be exploited that this functional is a supermodular [Fuj05] set-function w.r.t. the set  $\mathbb{S}$ .

**Supermodularity:** On a finite set  $\mathbb{V}$ , a set-function  $s : 2^{\mathbb{V}} \rightarrow \mathbb{R}^1$  is supermodular if for all  $\mathbb{A}, \mathbb{B} \subseteq \mathbb{V}$

$$s(\mathbb{A}) + s(\mathbb{B}) \leq s(\mathbb{A} \cap \mathbb{B}) + s(\mathbb{A} \cup \mathbb{B}) \quad (4.2)$$

holds. Modular, in this sense, means that subsets  $\mathbb{A}$  and  $\mathbb{B}$  of  $\mathbb{V}$  represent parts of a complex system and compositions of those modules expressed by union or intersection as well. Super indicates that if measuring these sets with  $s(\cdot)$ , the sum of the compositions (i.e., the right-hand side) is larger than the sum of the modules  $\mathbb{A}$  and  $\mathbb{B}$  themselves. In (4.1), the compression rates  $I(\mathcal{Y}_{\mathbb{S}}; \mathcal{Z}_{\mathbb{S}}|\mathcal{Z}_{\overline{\mathbb{S}}}, \mathcal{Q})$  are set-functions w.r.t. the set  $\mathbb{S}$ . Since the relevant information  $I(\mathcal{X}; \mathcal{Z}|\mathcal{Q})$  does not depend on  $\mathbb{S}$  and is constant in this regard, it can be omitted for showing the supermodularity. Applying the general definition for supermodular functions in (4.2) on the compression rates  $I(\mathcal{Y}_{\mathbb{S}}; \mathcal{Z}_{\mathbb{S}}|\mathcal{Z}_{\overline{\mathbb{S}}}, \mathcal{Q})$  it can be shown that for  $\mathbb{A}, \mathbb{B} \subseteq \mathbb{S}$

$$\begin{aligned} s(\mathbb{A}) + s(\mathbb{B}) &= I(\mathcal{Y}_{\mathbb{A}}; \mathcal{Z}_{\mathbb{A}}|\mathcal{Z}_{\overline{\mathbb{A}}}, \mathcal{Q}) + I(\mathcal{Y}_{\mathbb{B}}; \mathcal{Z}_{\mathbb{B}}|\mathcal{Z}_{\overline{\mathbb{B}}}, \mathcal{Q}) \\ &\leq I(\mathcal{Y}_{\mathbb{A} \cup \mathbb{B}}; \mathcal{Z}_{\mathbb{A} \cup \mathbb{B}}|\mathcal{Z}_{\overline{\mathbb{A} \cup \mathbb{B}}}, \mathcal{Q}) + I(\mathcal{Y}_{\mathbb{A} \cap \mathbb{B}}; \mathcal{Z}_{\mathbb{A} \cap \mathbb{B}}|\mathcal{Z}_{\overline{\mathbb{A} \cap \mathbb{B}}}, \mathcal{Q}) \\ &= s(\mathbb{A} \cup \mathbb{B}) + s(\mathbb{A} \cap \mathbb{B}) \end{aligned} \quad (4.3)$$

holds. The proof for this originates from [CW14] and is given in Appendix A in more detail.

**A Greedy optimization structure:** The supermodularity can be exploited in order to create an optimization structure. According to [McC05], a greedy algorithm can be applied to find the extreme points of the solution space of a supermodular function. Following the argumentation in [CW14], the rate constraints

$$I(\mathcal{Y}_{\mathbb{S}}; \mathcal{Z}_{\mathbb{S}}|\mathcal{Z}_{\overline{\mathbb{S}}}, \mathcal{Q}) \leq \sum_{m \in \mathbb{S}} C_m$$

of the inner bound in (3.40) define the solution space described by the polytope  $\mathcal{P}_D$ . For two sets  $\mathbb{S}_j \subseteq \{1, \dots, M\}$  and  $\mathbb{S}_{j-1} \subset \mathbb{S}_j$  with  $\mathbb{S}_j \setminus \mathbb{S}_{j-1} = \{j\}$  and their complements  $\overline{\mathbb{S}}_j = \{1, \dots, M\} \setminus \mathbb{S}_j$  and  $\overline{\mathbb{S}}_{j-1} = \{1, \dots, M\} \setminus \mathbb{S}_{j-1}$ , it can be seen that

$$I(\mathcal{Y}_{\mathbb{S}_j}; \mathcal{Z}_{\mathbb{S}_j}|\mathcal{Z}_{\overline{\mathbb{S}}_j}, \mathcal{Q}) - I(\mathcal{Y}_{\mathbb{S}_{j-1}}; \mathcal{Z}_{\mathbb{S}_{j-1}}|\mathcal{Z}_{\overline{\mathbb{S}}_{j-1}}, \mathcal{Q}) \geq 0 \quad (4.4)$$

<sup>1</sup>The notation  $2^{\mathbb{V}}$  represents the power set including all subsets of  $\mathbb{V}$ .

holds, since the additional element  $j$  in  $\mathbb{S}_j$  compared to  $\mathbb{S}_{j-1}$  can only increase the compression rate  $I(\mathcal{Y}_{\mathbb{S}_j}; \mathcal{Z}_{\mathbb{S}_j} | \mathcal{Z}_{\bar{\mathbb{S}}_j}, \mathcal{Q})$ . Combining (3.40) with (4.4) delivers

$$\begin{aligned}
 I(\mathcal{Y}_{\mathbb{S}_j}; \mathcal{Z}_{\mathbb{S}_j} | \mathcal{Z}_{\bar{\mathbb{S}}_j}, \mathcal{Q}) - I(\mathcal{Y}_{\mathbb{S}_{j-1}}; \mathcal{Z}_{\mathbb{S}_{j-1}} | \mathcal{Z}_{\bar{\mathbb{S}}_{j-1}}, \mathcal{Q}) &\leq \sum_{m \in \mathbb{S}_j} C_m - \sum_{m \in \mathbb{S}_{j-1}} C_m \\
 \mathbb{E}_{\mathcal{Y}_{\mathbb{S}_j}, \mathcal{Z}, \mathcal{Q}} \left[ \log_2 \frac{p(\mathbf{z}_{\mathbb{S}_j} | \mathbf{y}_{\mathbb{S}_j}, \mathbf{z}_{\bar{\mathbb{S}}_j}, \mathbf{q})}{p(\mathbf{z}_{\mathbb{S}_j} | \mathbf{z}_{\bar{\mathbb{S}}_j}, \mathbf{q})} \right] - \mathbb{E}_{\mathcal{Y}_{\mathbb{S}_{j-1}}, \mathcal{Z}, \mathcal{Q}} \left[ \log_2 \frac{p(\mathbf{z}_{\mathbb{S}_{j-1}} | \mathbf{y}_{\mathbb{S}_{j-1}}, \mathbf{z}_{\bar{\mathbb{S}}_{j-1}}, \mathbf{q})}{p(\mathbf{z}_{\mathbb{S}_{j-1}} | \mathbf{z}_{\bar{\mathbb{S}}_{j-1}}, \mathbf{q})} \right] &\leq C_j \\
 \mathbb{E}_{\mathcal{Y}_{\mathbb{S}_j}, \mathcal{Z}, \mathcal{Q}} \left[ \log_2 \frac{p(\mathbf{z}_{\mathbb{S}_j} | \mathbf{y}_{\mathbb{S}_j}, \mathbf{q})}{p(\mathbf{z}_{\mathbb{S}_j} | \mathbf{z}_{\bar{\mathbb{S}}_j}, \mathbf{q})} \right] - \mathbb{E}_{\mathcal{Y}_{\mathbb{S}_{j-1}}, \mathcal{Z}, \mathcal{Q}} \left[ \log_2 \frac{p(\mathbf{z}_{\mathbb{S}_{j-1}} | \mathbf{y}_{\mathbb{S}_{j-1}}, \mathbf{q})}{p(\mathbf{z}_{\mathbb{S}_{j-1}} | \mathbf{z}_{\bar{\mathbb{S}}_{j-1}}, \mathbf{q})} \right] &\leq C_j \quad (4.5)
 \end{aligned}$$

$$\mathbb{E}_{\mathcal{Y}_{\mathbb{S}_j}, \mathcal{Z}, \mathcal{Q}} \left[ \log_2 \frac{\prod_{i \in \mathbb{S}_j} p(z_i | y_i, \mathbf{q})}{p(\mathbf{z}_{\mathbb{S}_j} | \mathbf{z}_{\bar{\mathbb{S}}_j}, \mathbf{q})} \right] - \mathbb{E}_{\mathcal{Y}_{\mathbb{S}_{j-1}}, \mathcal{Z}, \mathcal{Q}} \left[ \log_2 \frac{\prod_{i \in \mathbb{S}_{j-1}} p(z_i | y_i, \mathbf{q})}{p(\mathbf{z}_{\mathbb{S}_{j-1}} | \mathbf{z}_{\bar{\mathbb{S}}_{j-1}}, \mathbf{q})} \right] \leq C_j \quad (4.6)$$

$$\mathbb{E}_{\mathcal{Y}_{\mathbb{S}_j}, \mathcal{Z}, \mathcal{Q}} \left[ \log_2 \frac{\prod_{i \in \mathbb{S}_j} p(z_i | y_i, \mathbf{q})}{p(\mathbf{z}_{\mathbb{S}_j} | \mathbf{z}_{\bar{\mathbb{S}}_j}, \mathbf{q})} \frac{p(\mathbf{z}_{\mathbb{S}_{j-1}} | \mathbf{z}_{\bar{\mathbb{S}}_{j-1}}, \mathbf{q})}{\prod_{i \in \mathbb{S}_{j-1}} p(z_i | y_i, \mathbf{q})} \right] \leq C_j$$

$$\mathbb{E}_{\mathcal{Y}_j, \mathcal{Z}, \mathcal{Q}} \left[ \log_2 \frac{p(z_j | y_j, \mathbf{q}) p(\mathbf{z}_{\mathbb{S}_{j-1}} | \mathbf{z}_{\bar{\mathbb{S}}_{j-1}}, \mathbf{q})}{p(\mathbf{z}_{\mathbb{S}_j} | \mathbf{z}_{\bar{\mathbb{S}}_j}, \mathbf{q})} \right] \leq C_j$$

$$\mathbb{E}_{\mathcal{Y}_j, \mathcal{Z}_j, \mathbf{z}_{\bar{\mathbb{S}}_j}, \mathcal{Q}} \left[ \log_2 \frac{p(z_j | y_j, \mathbf{q})}{p(z_j | \mathbf{z}_{\bar{\mathbb{S}}_j}, \mathbf{q})} \right] \leq C_j$$

$$\mathbb{E}_{\mathcal{Y}_j, \mathcal{Z}_j, \mathbf{z}_{\bar{\mathbb{S}}_j}, \mathcal{Q}} \left[ \log_2 \frac{p(z_j | y_j, \mathbf{z}_{\bar{\mathbb{S}}_j}, \mathbf{q})}{p(z_j | \mathbf{z}_{\bar{\mathbb{S}}_j}, \mathbf{q})} \right] \leq C_j \quad (4.7)$$

$$I(\mathcal{Y}_j; \mathcal{Z}_j | \mathcal{Z}_{\bar{\mathbb{S}}_j}, \mathcal{Q}) \leq C_j. \quad (4.8)$$

Equation (4.5) holds due to the Markov property given in (3.39). Moreover, this Markov property ensures that different  $z \in \mathbb{S}_j$  are independent given  $\mathbf{y}_{\mathbb{S}_j}$  and different  $z \in \mathbb{S}_{j-1}$  are independent given  $\mathbf{y}_{\mathbb{S}_{j-1}}$  resulting in (4.6). Finally, the Markov property in (3.39) allows to extend the pmf  $p(z_j | y_j, \mathbf{q})$  to  $p(z_j | y_j, \mathbf{z}_{\bar{\mathbb{S}}_j}, \mathbf{q})$  resulting in equation (4.7). The left part of (4.8) can be interpreted as the maximum rate  $R_j$  with which sensor  $j$  can transmit information, resulting in

$$R_j = I(\mathcal{Y}_j; \mathcal{Z}_j | \mathcal{Z}_{\bar{\mathbb{S}}_j}, \mathcal{Q}). \quad (4.9)$$

In order to examine an optimization structure, an example with three sensors is considered resulting in three compression rates, one for each sensor.

$$\begin{aligned}
 \mathbb{S}_j &= \{1\} & \mathbb{S}_{j-1} &= \{\} & j &= 1 & R_1 &= I(\mathcal{Y}_1; \mathcal{Z}_1 | \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Q}) \\
 \mathbb{S}_j &= \{1, 2\} & \mathbb{S}_{j-1} &= \{1\} & j &= 2 & R_2 &= I(\mathcal{Y}_2; \mathcal{Z}_2 | \mathcal{Z}_3, \mathcal{Q}) \\
 \mathbb{S}_j &= \{1, 2, 3\} & \mathbb{S}_{j-1} &= \{1, 2\} & j &= 3 & R_3 &= I(\mathcal{Y}_3; \mathcal{Z}_3 | \mathcal{Q})
 \end{aligned} \quad (4.10)$$

Observing the three rates in (4.10) suggests a greedy optimization in reverse order, i.e., starting with the third sensor and optimizing the first sensor at last. In this way, sensors being currently optimized can exploit the quantizer mappings of previously designed

sensors by Wyner-Ziv coding. However, for the same three-sensor scenario, the emerging rates could look quite different depending on the sets  $\mathbb{S}_j$  and  $\mathbb{S}_{j-1}$ .

$$\begin{aligned}
 \mathbb{S}_j &= \{1\} & \mathbb{S}_{j-1} &= \{\} & j &= 1 & R_1 &= I(\mathcal{Y}_1; \mathcal{Z}_1 | \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Q}) \\
 \mathbb{S}_j &= \{1, 3, 2\} & \mathbb{S}_{j-1} &= \{1, 3\} & j &= 2 & R_2 &= I(\mathcal{Y}_2; \mathcal{Z}_2 | \mathcal{Q}) \\
 \mathbb{S}_j &= \{1, 3\} & \mathbb{S}_{j-1} &= \{1\} & j &= 3 & R_3 &= I(\mathcal{Y}_3; \mathcal{Z}_3 | \mathcal{Z}_2, \mathcal{Q})
 \end{aligned} \tag{4.11}$$

The rates in (4.11) suggest an optimization order starting with the second sensor, followed by the third sensor, and optimizing the first sensor at last. Note that each optimization order results in a different extreme point of the solution space  $\mathcal{P}_D$ . In order to find all extreme points of  $\mathcal{P}_D$ , all  $M!$  different optimization orders have to be considered. In addition, it is generally not known in advance which extreme point of  $\mathcal{P}_D$  is the best. Exploiting this greedy optimization structure, the optimization problem for any optimization order<sup>2</sup>  $\pi_1 \prec \pi_2 \prec \dots \prec \pi_M$  can be reformulated to

$$\begin{aligned}
 \max_{\mathbb{P}} I(\mathcal{X}; \mathcal{Z} | \mathcal{Q}) \text{ s.t. } & I(\mathcal{Y}_{\pi_m}; \mathcal{Z}_{\pi_m} | \mathcal{Z}_{\tilde{\mathbb{S}}_{< m}}, \mathcal{Q}) \leq C_{\pi_m} \\
 & \forall m \in \{1, 2, \dots, M\}
 \end{aligned} \tag{4.12}$$

with  $\pi_m$  representing the  $m$ -th position in the optimization order and the set  $\tilde{\mathbb{S}}_{< m}$  corresponding to  $\tilde{\mathbb{S}}_{< m} = \{\pi_1, \dots, \pi_{m-1}\}$ .<sup>3</sup> For notational simplicity, all following equations will assume a natural optimization order and omit other permutations. The complete convex hull of the solution space can be determined by convex combination [BV04; HL04] of all extreme points using time-sharing with parameter  $\mathcal{Q}$ , which is illustrated for a two-sensor scenario in Subsection 4.2.2. However, this thesis does not focus on characterizing the complete rate region but on optimal solutions located in the extreme points. Therefore, the time-sharing parameter is omitted in the following equations. The optimization problem in (4.12) can be solved by the method of Lagrangian multipliers resulting in

$$L_{\text{GDIB}} = I(\mathcal{X}; \mathcal{Z}) - \sum_{m=1}^M \beta_m \cdot I(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{< m}). \tag{4.13}$$

Applying the chain rule of mutual information to  $I(\mathcal{X}; \mathcal{Z})$ , the optimization approach becomes

$$L_{\text{GDIB}} = \sum_{m=1}^M I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{< m}) - \beta_m \cdot I(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{< m}). \tag{4.14}$$

<sup>2</sup>The notation  $\pi_1 \prec \pi_2 \prec \dots \prec \pi_M$  indicates that the sensor on position  $\pi_1$  is optimized first, followed by the sensor on position  $\pi_2$ . The last sensor optimized is the sensor on position  $\pi_M$ .

<sup>3</sup>Example: 3 sensors with optimization order  $\pi_1 = 2 \prec \pi_2 = 3 \prec \pi_3 = 1$  leading to  $\tilde{\mathbb{S}}_{< 1} = \{\}$ ,  $\tilde{\mathbb{S}}_{< 2} = \{2\}$  and  $\tilde{\mathbb{S}}_{< 3} = \{2, 3\}$ . The compression rates result in  $I(\mathcal{Y}_2; \mathcal{Z}_2 | \mathcal{Q})$ ,  $I(\mathcal{Y}_3; \mathcal{Z}_3 | \mathcal{Z}_2, \mathcal{Q})$  and  $I(\mathcal{Y}_1; \mathcal{Z}_1 | \mathcal{Z}_2, \mathcal{Z}_3, \mathcal{Q})$ , respectively.

It is obvious that by pursuing the greedy optimization approach, equation (4.14) can be decomposed into  $M$  optimization problems

$$L_{\text{GDIB}}^{(1)} = I(\mathcal{X}; \mathcal{Z}_1) - \beta_1 I(\mathcal{Y}_1; \mathcal{Z}_1) \quad (4.15a)$$

$$\vdots$$

$$L_{\text{GDIB}}^{(M)} = I(\mathcal{X}; \mathcal{Z}_M | \mathcal{Z}_{<M}) - \beta_M I(\mathcal{Y}_M; \mathcal{Z}_M | \mathcal{Z}_{<M}). \quad (4.15b)$$

Equations (4.15a) to (4.15b) emphasize the greedy optimization structure. To be more specific, the first sensor is optimized by maximizing (4.15a), which resembles the original scalar IB optimization. Subsequent sensors are optimized by exploiting the quantizer mappings of all previously designed sensors by applying the Wyner-Ziv coding principle. In this way, all quantizers are jointly optimized in a successive way. Note that no information has to be exchanged during run-time. This greedy approach assumes that the common decoder has already received information from previous sensors such that only the remaining part has to be forwarded. As in the original IB method, the Lagrange multipliers  $\beta_m$  serve as a trade-off parameter between the preservation of relevant information and compression. Each  $\beta_m$  has to be adjusted such that the rate constraint  $I(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{<m}) \leq C_m$  is fulfilled.

## 4.2 An Algorithmic Solution

This section introduces an algorithmic solution to solve the optimization problems given in (4.15a) to (4.15b). These objectives can be solved by equating the derivative w.r.t. the mapping  $p(z_m | y_m)$  to zero. This results in the update equation for sensor  $m$

$$p(z_m | y_m) = \frac{e^{-d_{\beta_m}(y_m, z_m)}}{\sum_{z_m} e^{-d_{\beta_m}(y_m, z_m)}} \quad (4.16)$$

with the exponent  $d_{\beta_m}(y_m, z_m)$  given as

$$d_{\beta_m}(y_m, z_m) := \mathbb{E}_{\mathbf{z}_{<m} | y_m} \left[ \frac{1}{\beta_m} \cdot D_{\text{KL}} [p(x | y_m, \mathbf{z}_{<m}) || p(x | \mathbf{z}_{\leq m})] - \log p(z_m | \mathbf{z}_{<m}) \right]. \quad (4.17)$$

Note that update equation in (4.16) is an implicit equation, since  $p(z_m | \mathbf{z}_{<m})$  and  $p(x | \mathbf{z}_{\leq m})$  also depend on the mapping  $p(z_m | y_m)$ . As in the original IB method, (4.16) can be solved using an extension of the Blahut-Arimoto algorithm. A detailed derivation of this solution is given in Appendix B.

The extended Blahut-Arimoto algorithm that solves the optimization problem for sensor  $m$  and a specific  $\beta_m$  is given in Algorithm 4. Lines 2 and 3 calculate required pmfs, which do not depend on  $z_m$  and so do not change in the loop. In lines 6 and 7, the joint pmf  $p(\mathbf{z}_{\leq m}, y_m, x)$  is calculated, which is the basis of all subsequent equations. Using this pmf, the KL divergence  $D_{\text{KL}}(y_m, \mathbf{z}_{\leq m})$  and the statistical distance measure  $d_{\beta_m}(z_m, y_m)$

---

**Algorithm 4:** Extended Blahut-Arimoto algorithm
 

---

```

input      :  $m, p(y_m, x), p^{\text{init}}(z_m|y_m), p(\mathbf{z}_{<m}|x), \beta_m, \epsilon$ 
output    :  $p(z_m|y_m) \in [0, 1]$ 
1 begin
   initialization:
        $p(z_m|y_m)^{(0)} \leftarrow p^{\text{init}}(z_m|y_m),$ 
        $l \leftarrow 1$ 
2    $p(\mathbf{z}_{<m}, y_m, x) = p(\mathbf{z}_{<m}|x)p(y_m, x)$ 
3    $p(x|\mathbf{z}_{<m}, y_m) = p(\mathbf{z}_{<m}, y_m, x) / \sum_x p(\mathbf{z}_{<m}, y_m, x)$ 
4    $p(\mathbf{z}_{<m}|y_m) = \sum_x p(\mathbf{z}_{<m}, y_m, x) / \sum_x p(y_m, x)$ 
5   do
6       // calculate  $p(\mathbf{z}_{\leq m}, y_m, x)$ 
7        $p(\mathbf{z}_{\leq m}, y_m, x)^{(l)} = p(\mathbf{z}_{<m}, y_m, x)p(z_m|y_m)^{(l-1)}$ 
8       // KL divergence  $D_{\text{KL}}(y_m, \mathbf{z}_{\leq m})$  of (4.17)
9        $p(x|\mathbf{z}_{\leq m})^{(l)} = \sum_{y_m} p(\mathbf{z}_{\leq m}, y_m, x)^{(l)} / \sum_{y_m, x} p(\mathbf{z}_{\leq m}, y_m, x)^{(l)}$ 
10       $D_{\text{KL}}(y_m, \mathbf{z}_{\leq m})^{(l)} = \sum_x p(x|\mathbf{z}_{<m}, y_m) \cdot \log \frac{p(x|\mathbf{z}_{<m}, y_m)}{p(x|\mathbf{z}_{\leq m})^{(l)}}$ 
11      // distance  $d_{\beta_m}(z_m, y_m)$  (4.17)
12       $p(z_m|\mathbf{z}_{<m})^{(l)} = \sum_{y_m, x} p(\mathbf{z}_{\leq m}, y_m, x)^{(l)} / \sum_{y_m, x} p(\mathbf{z}_{<m}, y_m, x)$ 
13       $d_{\beta_m}(z_m, y_m)^{(l)} = \sum_{\mathbf{z}_{<m}} p(\mathbf{z}_{<m}|y_m) \cdot \left[ \frac{1}{\beta_m} D_{\text{KL}}(y_m, \mathbf{z}_{\leq m})^{(l)} - \log p(z_m|\mathbf{z}_{<m})^{(l)} \right]$ 
14      // update quantizer  $p(z_m|y_m)$ 
15       $p(z_m|y_m)^{(l)} = \frac{1}{\sum_z e^{-d_{\beta_m}(z_m, y_m)^{(l)}}} e^{-d_{\beta_m}(z_m, y_m)^{(l)}}$ 
16       $l \leftarrow l + 1$ 
17 while  $D_{\text{JS}}[p^{(l)}(z_m|y_m) || p^{(l-1)}(z_m|y_m)] > \epsilon$ 

```

---

can be calculated in lines 8-10 and 11-13, respectively. Using  $d_{\beta_m}(z_m, y_m)$  the quantizer mapping  $p(z_m|y_m)$  can be updated. This procedure is done until a convergence criterion is fulfilled, e.g., the JS divergence between successive quantizer mappings does not change significantly anymore. In particular,  $D_{\text{JS}}[p^{(l)}(z_m|y_m) || p^{(l-1)}(z_m|y_m)] \leq \epsilon$  shall be smaller than or equal to a predefined accuracy parameter  $\epsilon$ .

A flowchart of the overall GDIB algorithm is given in Figure 4.1. The previously described extended Blahut-Arimoto algorithm is performed in the block "ext. BA algorithm". More precisely, it computes (4.16) with (4.17) for a specific sensor  $m$  and a specific  $\beta_m$ . Note that the extended Blahut-Arimoto algorithm itself contains a loop, which iteratively optimizes the quantizer mapping  $p(z_m|y_m)$  of sensor  $m$ . Since the compression rate  $I(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{<m})$  is a monotonic increasing function in  $\beta_m$  the rate-fulfilling Lagrange multiplier  $\beta_m$  can be determined using a bisection search, such that  $I(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{<m}) \leq C_m$ . This is performed in the "loop to adjust  $\beta$ ". Of course, all this has to be done for all sensors in the "loop over sensors". Once all quantizers are optimized, the algorithm ends. In general, this procedure needs to be done for all permutations of the optimization order, which is not considered in Figure 4.1.

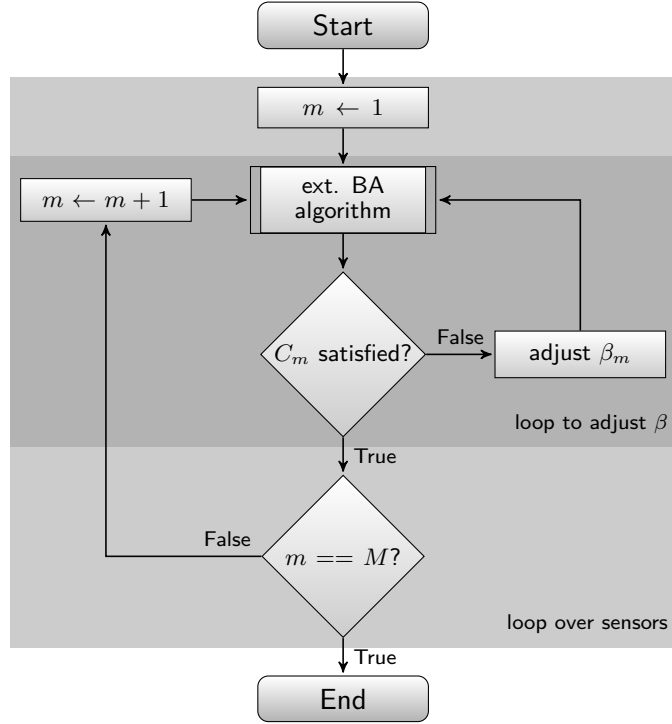


Figure 4.1: Flowchart of the overall GDIB algorithm

### 4.2.1 Simulation Setup

In the following subsections, the performance of the proposed GDIB algorithm is investigated in different non-cooperative CEO scenarios. The relevant signal is corrupted by additive white Gaussian measurement noise at each sensor defined by the SNR  $\gamma_m$ . In order to be able to perform the optimization with the GDIB algorithm, the measurements are uniformly pre-quantized with  $|\mathbb{Y}_m| = 64$  bins. The optimization algorithm is initialized by a uniform quantization. This initialization results in coherent output clusters and generally leads to a good performance.

There are two types of scenarios that are investigated, symmetric and asymmetric scenarios. Symmetric, in this case, indicates that each sensor has the same SNR,  $|\mathbb{Z}_m|$ , and capacity constraints  $C_m$ . Whereas, this is not the case for asymmetric scenarios. The advantage of symmetric scenarios is that the optimization order does not play any role. Therefore, it is not necessary to perform the GDIB optimization for all possible optimization orders. This highly reduces the computational complexity.

### 4.2.2 Achievable Rate Tuples

Applying the GDIB algorithm for a symmetric  $M = 2$  sensor scenario, the achievable rate tuples can be illustrated, as given in Figure 4.2. The rate tuples  $(R_1, R_2)$  are depicted for specific distortions, i.e., for specific relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$ . Note that the rate tuples either correspond to  $(I(\mathcal{Y}_1; \mathcal{Z}_1), I(\mathcal{Y}_2; \mathcal{Z}_2|\mathcal{Z}_1))$  or  $(I(\mathcal{Y}_1; \mathcal{Z}_1|\mathcal{Z}_2), I(\mathcal{Y}_2; \mathcal{Z}_2))$  depending on the optimization order. Time-sharing between both optimization orders

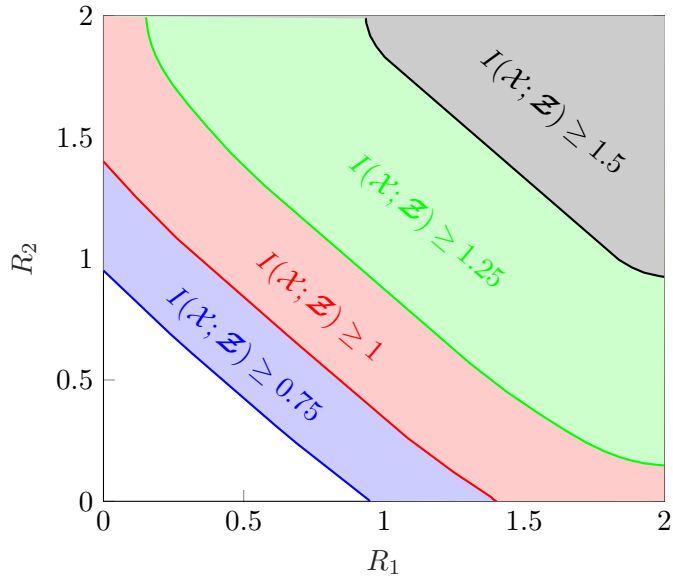


Figure 4.2: Achievable rate tuples for different relevant mutual information using the GDIB algorithm in a symmetric scenario with two sensors,  $|\mathbb{X}| = 4$ ,  $|\mathbb{Z}_m| = 4$  and SNRs  $\gamma_m = 8$  dB

delivers the convex regions depicted in Figure 4.2. The relevant signal is taken from a uniformly distributed 4-ASK alphabet while the SNRs are chosen to be  $\gamma_m = 8$  dB.

The different colored regions correspond to particular thresholds for the relevant information  $I(\mathcal{X}; \mathcal{Z}) \geq \{0.75, 1, 1.25, 1.5\}$ . Note that on the left border of a specific region, equality to the threshold holds. For low values of  $I(\mathcal{X}; \mathcal{Z}) \lesssim 1.15$ , the relevant information can be achieved by a single sensor only. Here, the rate of one sensor can become zero, while  $I(\mathcal{X}; \mathcal{Z})$  is completely provided by the other sensor. Larger relevant information requires both sensors to contribute.

### 4.2.3 Contribution of Wyner-Ziv coding

The GDIB approach applies the Wyner-Ziv coding principle by exploiting the mappings of previously designed quantizers. In this sense, the compression of a sensor can be optimized, assuming that the common receiver has already received information from the previous sensors. In other words, a current sensor only needs to transmit data that the receiver has not seen so far. To analyze this behavior, Figure 4.3 illustrates the relevance compression plane for a symmetric two-sensor scenario with a signal-to-noise ratio of  $\gamma_m = 8$  dB and a 4-ASK relevant signal. In particular, the amount of information the second sensor can contribute is depicted for different optimizations of the first sensor. The black dashed line represents the relevance compression curve for the first sensor obtained by varying  $\beta_1$ , i.e., it depicts  $I(\mathcal{X}; \mathcal{Z}_1)$  vs.  $I(\mathcal{Y}_1; \mathcal{Z}_1)$ . The colored curves represent the contribution of the second sensor  $I(\mathcal{X}; \mathcal{Z}_2 | \mathcal{Z}_1)$  vs.  $I(\mathcal{Y}_2; \mathcal{Z}_2 | \mathcal{Z}_1)$ , after optimizing the first one with specific values for  $\beta_1$ . The corresponding points  $(I(\mathcal{X}; \mathcal{Z}_1), I(\mathcal{Y}_1; \mathcal{Z}_1))$  are represented by the colored squared markers on the black dashed curve. Note that the overall performance of the network generally results from the sum of both sensors, which is not depicted here.



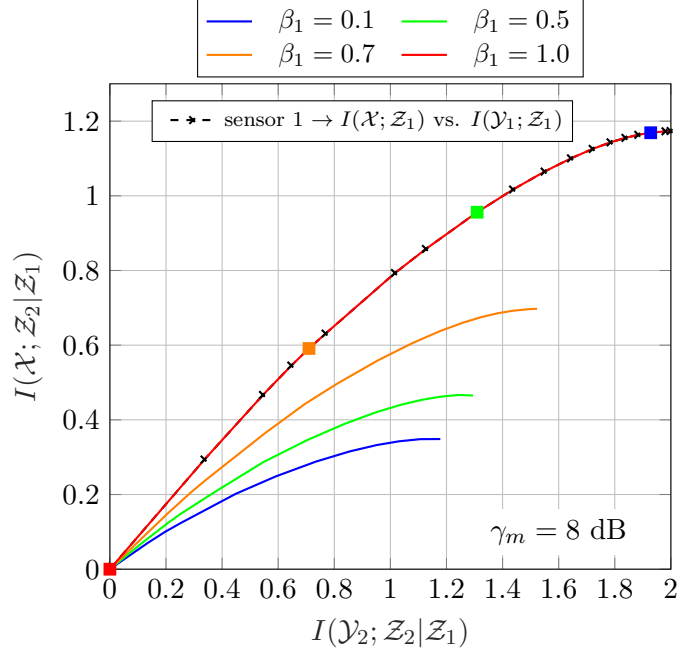


Figure 4.3: Contribution of the second sensor applying the GDIB algorithm in a symmetric scenario with two sensors,  $|\mathbb{X}| = 4$ ,  $|\mathbb{Z}_m| = 4$  and SNRs  $\gamma_m = 8$  dB

It can be observed that if the first sensor has a good forward channel and therefore is able to transmit a large amount of information, the amount the second sensor can contribute is quite low. Hence, the higher the colored squared marker on the relevant compression curve of the first sensor, the lower the slope and the maximum of  $I(\mathcal{X}; \mathbb{Z}_2|\mathbb{Z}_1)$  of the second sensor. Decreasing the performance of the first sensor results in a gain in performance of the second sensor. The red case represents the special case where the first sensor cannot transmit any information. In this case, the second sensor performs as if no first sensor is available. The red curve lies on top of the black dashed one. This shows that in this symmetric scenario, the individual performance of both sensors is also symmetric.

#### 4.2.4 Influence of Sum-Rate

Figure 4.4 illustrates the relevant mutual information versus the sum-rate for a symmetric scenario with  $M = 5$  sensors. The sum-rate  $C_{\text{sum}} = \sum_{m=1}^M C_m$  is fixed to a value independent of  $M$ . Consequently, the available capacity of each forward link becomes  $C_m = \frac{C_{\text{sum}}}{M}$ . This represents a scenario where all  $M$  sensors equally share a common medium in an orthogonal way and a round-robin fashion. The relevant signal is taken from a uniformly distributed 4-ASK alphabet. The green curve represents the result for a fcCEO scenario defined in Subsection 3.6.2 whereas the black curve represents the independent scalar IB optimization as defined in Subsection 3.6.1. The gray-shaded area represents the non-achievable region, since  $I(\mathcal{X}; \mathbb{Z})$  is always smaller or equal to  $I(\mathcal{X}; \mathbb{Y})$ . As expected, the fcCEO scenario performs best and approaches  $I(\mathcal{X}; \mathbb{Y})$  for larger sum-rates. It serves as an upper bound for the distributed sensing scenario. For a sum-rate of  $C_{\text{sum}} \geq 10$  bit/s/Hz independent scalar IB optimization achieves the same relevant infor-

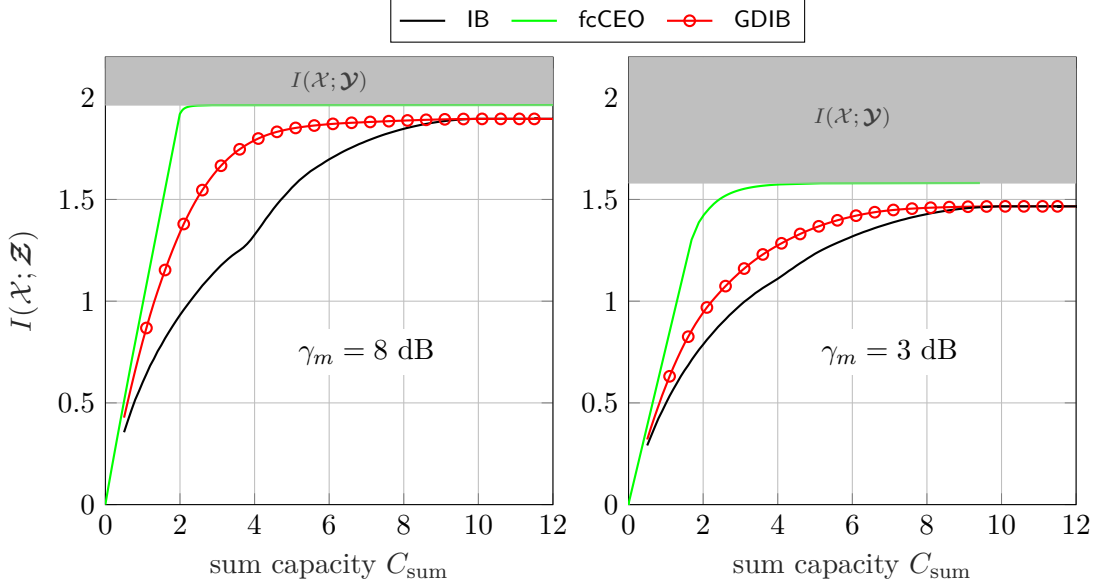


Figure 4.4: Relevant mutual information vs. sum capacity for a symmetric scenario with  $M = 5$  sensors,  $|\mathbb{X}| = 4$ , SNRs  $\gamma_m \in \{3, 8\}$  dB and  $|\mathbb{Z}_m| = 4$

mation as the GDIB approach. Naturally, this is caused by the fact that the performance of the GDIB approach saturates. In this area, no compression with stochastic mappings  $p(z_m|y_m)$  at the sensors is required. In the area of smaller sum-rates, the GDIB approach outperforms the independent scalar IB approach. The asymptotic loss of non-cooperative versus fully-cooperative distributed compression can be observed by the gap between the fcCEO scenario and the GDIB approach for very large sum-rates.

#### 4.2.5 Influence of the Network Size

In this subsection, the influence of the network size on the performance of the GDIB algorithm is investigated. First, a symmetric scenario where all sensors share the same medium in an orthogonal way and a round-robin fashion is considered, i.e., the sum-rate  $C_{\text{sum}}$  is fixed and equally distributed on all sensors  $C_m = \frac{C_{\text{sum}}}{M}$ . Consequently, the larger the network is, the smaller the individual link capacity of each sensor. Second, a symmetric scenario where each sensor has an independent link is considered. In this case, the sum-rate increases for larger networks  $C_{\text{sum}} = \sum_{m=1}^M C_m$  while the individual link capacity is the same for all sensors.

**Fixed Sum-Rate:** Figures 4.5 and 4.6 illustrate the influence of the network size onto the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  for sum-rates  $C_{\text{sum}} = 2.5$  bit/s/Hz and  $C_{\text{sum}} = 4.0$  bit/s/Hz, respectively. The relevant signal is taken from a uniformly distributed 4-ASK alphabet. The relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  cannot exceed  $I(\mathcal{X}; \mathcal{Y})$  due to the data-processing inequality. Hence, the gray-colored area represents the non-achievable region.

The black curves represent an independent scalar IB optimization, as introduced in Subsection 3.6.1. For higher SNRs  $\gamma_m = 8$  dB, this approach loses relevant information with

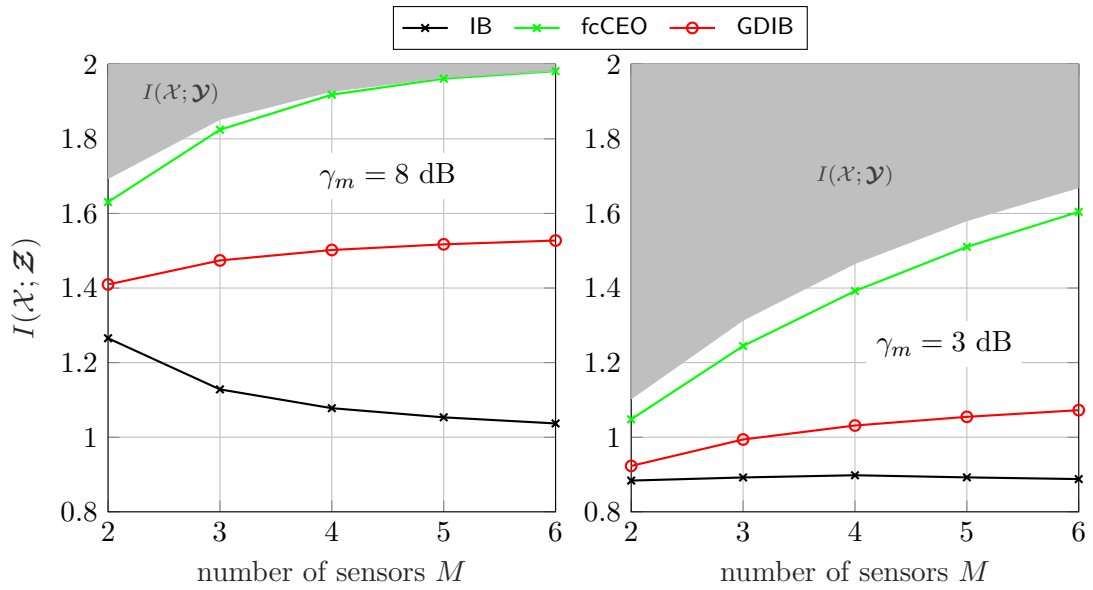


Figure 4.5: Relevant mutual information vs. number of sensors for fixed a sum-rate  $C_{\text{sum}} = 2.5$  bit/s/Hz,  $|\mathbb{X}| = 4$ ,  $|\mathbb{Z}_m| = 4$

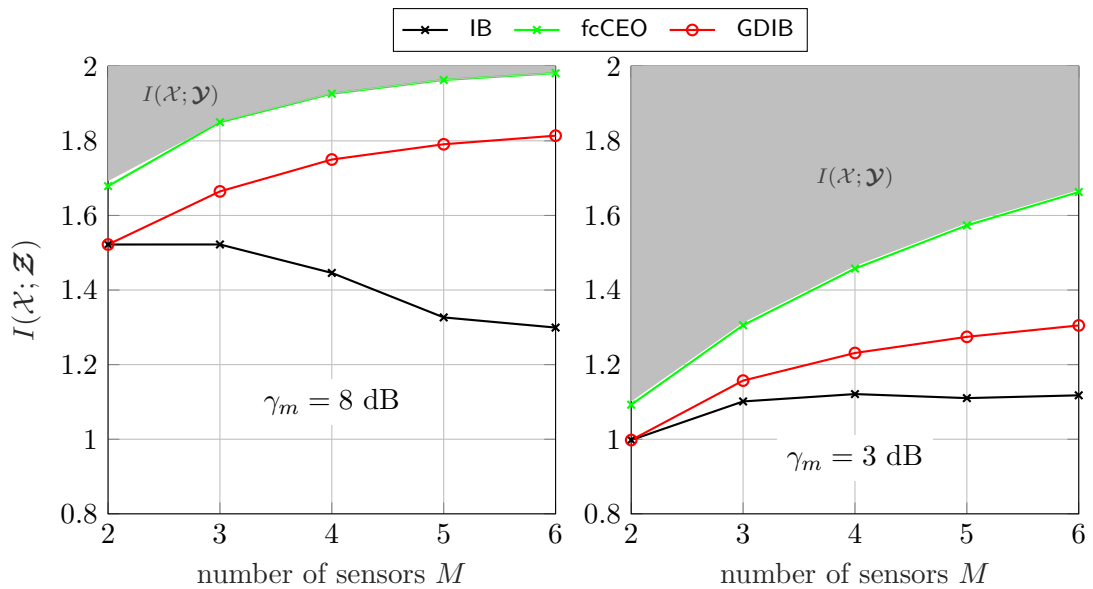


Figure 4.6: Relevant mutual information vs. number of sensors for fixed a sum-rate  $C_{\text{sum}} = 4.0$  bit/s/Hz,  $|\mathbb{X}| = 4$ ,  $|\mathbb{Z}_m| = 4$

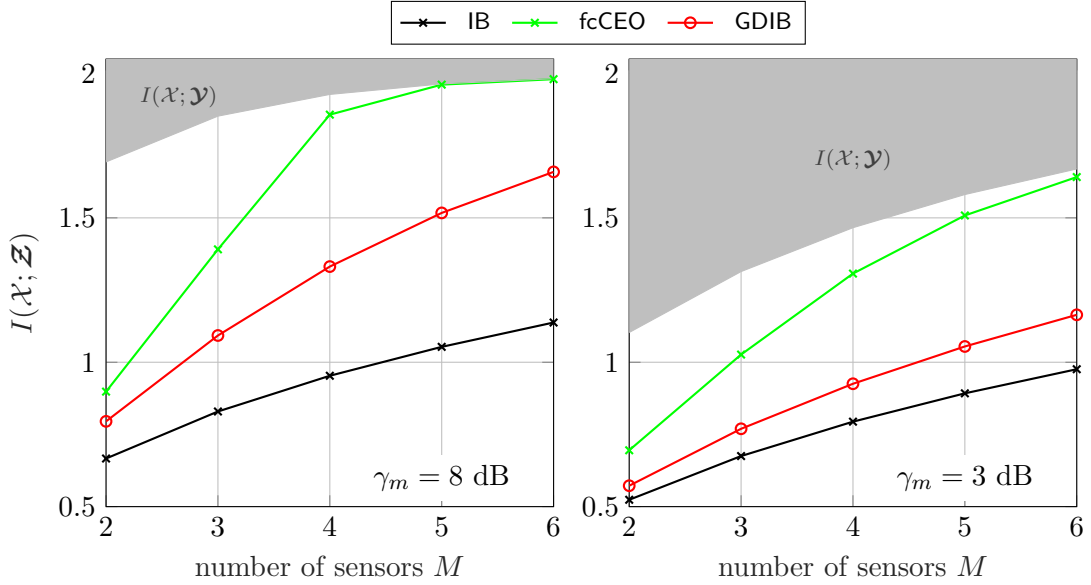


Figure 4.7: Relevant mutual information vs. number of sensors for fixed individual link capacity  $C_m = 0.5$  bit/s/Hz and  $|\mathbb{X}| = 4$ ,  $|\mathbb{Z}_m| = 4$

an increasing number of sensors. More precisely, increasing the number of sensors leads to smaller individual link capacities. Therefore, each sensor has to perform a stronger compression. It turns out that it is beneficial to use fewer sensors with higher compression rates. For lower SNRs of  $\gamma_m = 3$  dB, this degradation is not as pronounced for  $\gamma_m = 8$  dB. The red curves represent the GDIB approach. It becomes obvious that an increasing number of sensors does not lead to degradation but rather to an increase of relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$ . Therefore, using the GDIB algorithm for joint optimization of distributed quantizers leads to a significant gain compared to independent scalar IB optimization.

The green curve represents the result for a fcCEO scenario introduced in Subsection 3.6.2. Naturally, it clearly outperforms both other approaches. Of course, for larger network sizes, its performance is bounded by  $I(\mathcal{X}; \mathcal{Y})$ . The gap between the fcCEO scenario and the GDIB approach illustrates the limitation of non-cooperative distributed quantization, where no communication among the sensors is allowed. Comparing Figure 4.5 with Figure 4.6 reveals that for different sum-rates, the previously described effects are qualitatively similar.

**Increasing Sum-Rate:** So far, a fixed sum-rate  $C_{\text{sum}}$  distributed on all sensors has been considered. In the following, each sensor contributes a fixed link capacity  $C_m$  to the overall sum-rate  $C_{\text{sum}} = \sum_{m=1}^M C_m$ . Therefore, an increasing number of sensors will automatically increase the sum-rate  $C_{\text{sum}}$ . Figure 4.7 illustrates the influence of the network size onto the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  when each sensor contributes a link capacity of  $C_m = 0.5$  bit/s/Hz to the sum-rate. Again, the relevant signal is taken from a uniformly distributed 4-ASK alphabet. The gray-shaded area represents the non-achievable region since  $I(\mathcal{X}; \mathcal{Y})$  cannot be exceeded. The black curve represents the independent scalar IB

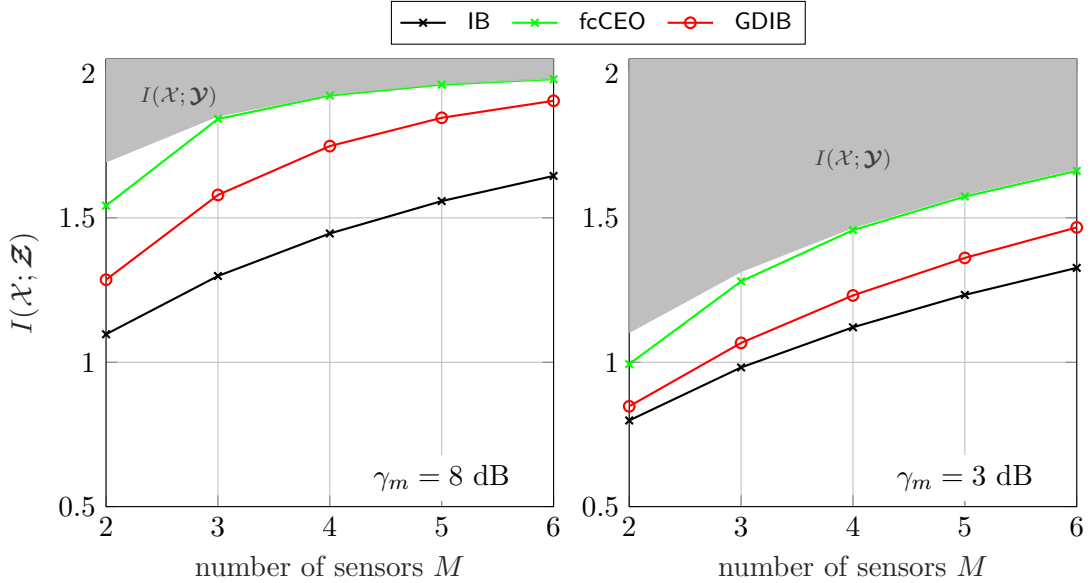


Figure 4.8: Relevant mutual information vs. number of sensors for fixed individual link capacity  $C_m = 1$  bit/s/Hz and  $|\mathbb{X}| = 4$ ,  $|\mathbb{Z}_m| = 4$

optimization from Subsection 3.6.1 while the red curve represents the GDIB approach. The green curve represents the performance of a fcCEO scenario. Since each additional sensor increases the available sum-rate, the overall performance of all approaches benefits from larger networks. Similar to the case with a fixed sum-rate, the GDIB approach outperforms independent scalar IB optimization since it exploits the Wyner-Ziv coding principle. This performance gap increases for larger network sizes due to increasing sum-rates. However, the gap to the fcCEO scenario observed for a fixed sum-rate in Figure 4.5 and 4.6 also occurs in Figure 4.7.

Figure 4.8 shows the performance for the same simulation but with a larger individual link capacity of  $C_m = 1.0$  bit/s/Hz. It can be observed that these larger individual link capacities generally produce qualitatively the same result as in Figure 4.7. The main difference is a better overall performance being closer to the upper bound  $I(\mathcal{X}; \mathcal{Y})$  in Figure 4.8.

#### 4.2.6 Influence of the Relevant Signal

**Gaussian Relevant Signal:** Figure 4.9 illustrates the relevant mutual information versus the sum-rate for the same scenario as in Figure 4.4 but for a Gaussian relevant signal  $x$  pre-quantized with  $|\mathbb{X}| = 64$  equidistant bins. Again, the green curve represents the fcCEO scenario defined in Subsection 3.6.2 whereas the black curve represents the independent scalar IB optimization defined in Subsection 3.6.1. In general, the Gaussian relevant signal does not change the quality of the result. The main difference is the amount of information available in the system, i.e.,  $I(\mathcal{X}; \mathcal{Y})$  increases compared to a 4-ASK signal. Although more information is available, the performance of the GDIB approach is nearly the same as in Figure 4.4. Therefore, the gap to the fcCEO scenario increases, especially

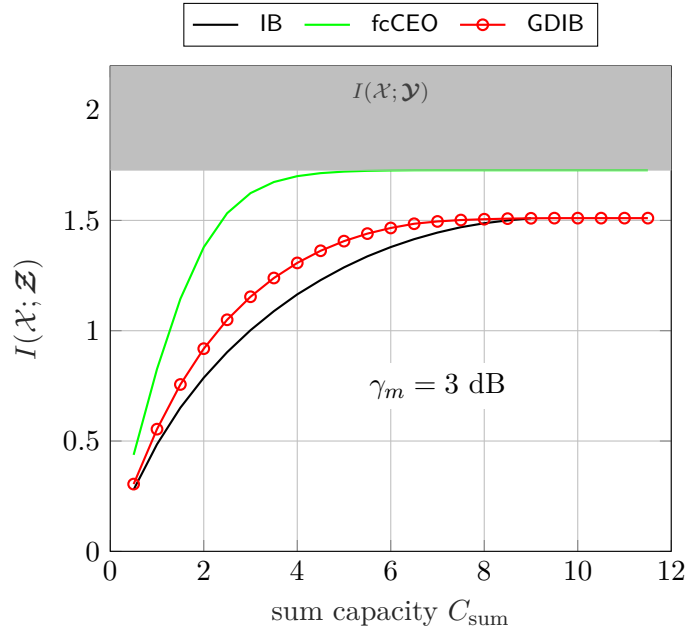


Figure 4.9: Relevant mutual information vs. sum capacity for a symmetric scenario with  $M = 5$  sensors, SNRs  $\gamma_m = 3$  dB,  $|\mathbb{Z}_m| = 4$  and a Gaussian relevant signal  $x$  with  $|\mathbb{X}| = 64$

for larger sum-rates. However, the GDIB approach still outperforms independent scalar IB optimization.

Figure 4.10 illustrates the influence of the network size onto the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  for a Gaussian relevant signal  $x$  pre-quantized with  $|\mathbb{X}| = 64$  bins. The simulation is done for sum-rates of  $C_{\text{sum}} = \{2.5, 4.0\}$  bit/s/Hz and an SNR  $\gamma_m = 3$  dB. Independent of the sum-rate, the GDIB algorithm outperforms independent scalar IB designed quantizers, which is shown in black. This can again be explained by the GDIB algorithm exploiting the mappings of previously designed quantizers. However, the simulation results reveal a large gap to the fcCEO scenario, where each sensor has access to all measurements in the network. In general, the result resembles the ones obtained for a 4-ASK relevant signal in Figure 4.5 and 4.6. Thus, from Figure 4.9 and 4.10, it can be concluded that the GDIB algorithm can also be applied for pre-quantized continuous relevant signals.

**Discrete Relevant Signals:** Figure 4.11 illustrates the influence of the network size onto the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  with a sum-rate of  $C_{\text{sum}} = 4.0$  bit/s/Hz for different relevant signals. All curves illustrate the performance using the GDIB algorithm. The relevant signal is taken from a uniformly distributed ASK alphabet where the cardinality  $|\mathbb{X}|$  determines the modulation order. The output cardinality is chosen to be  $|\mathbb{Z}_m| = 8$  for all modulation schemes. In general, apart from the 2-ASK curve for  $\gamma_m = 8$  dB, the curves look qualitatively the same. In all cases, the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  increases for larger network sizes. This confirms that the GDIB algorithm can be applied for arbitrary input cardinalities  $|\mathbb{X}|$ . Naturally, the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  in the case of a 2-ASK relevant signal is limited to one bit/s/Hz. Hence,

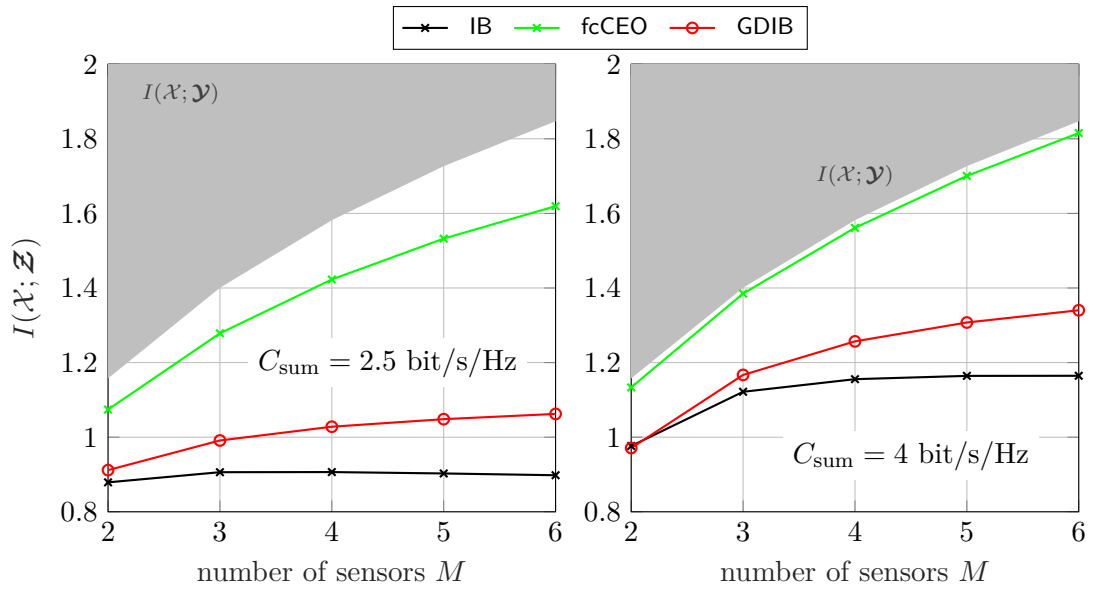


Figure 4.10: Relevant mutual information vs. number of sensors for a fixed sum-rate  $C_{\text{sum}} \in \{2.5, 4.0\}$  bit/s/Hz,  $|\mathbb{Z}_m| = 4$ ,  $\gamma_m = 3$  dB and a Gaussian relevant signal  $x$  with  $|\mathbb{X}| = 64$

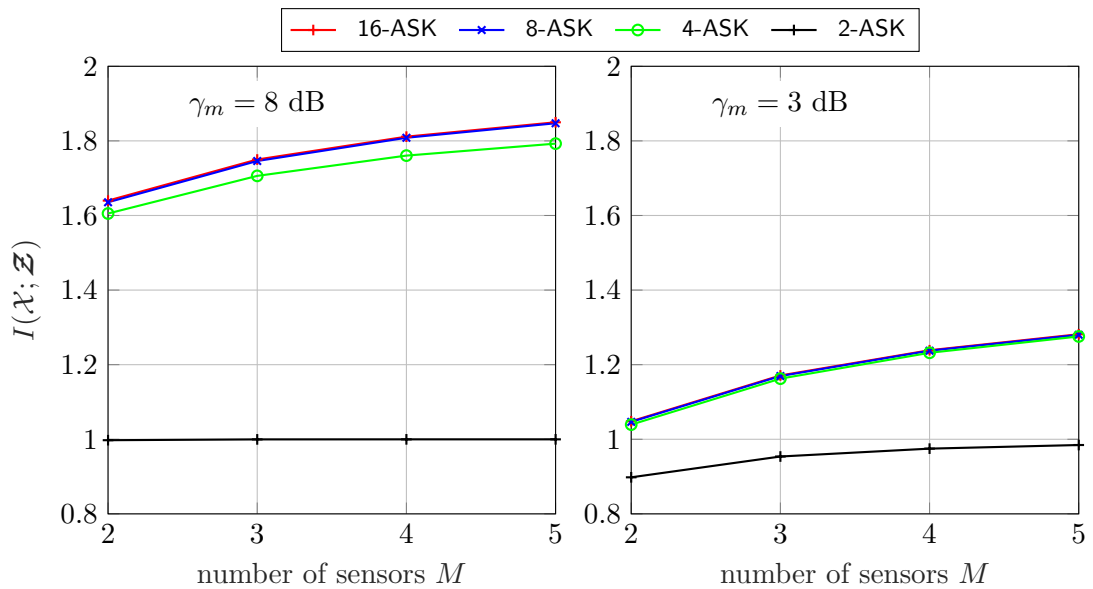


Figure 4.11: Relevant mutual information vs. number of sensors after GDIB optimization for a fixed sum-rate  $C_{\text{sum}} = 4.0$  bit/s/Hz,  $|\mathbb{Z}_m| = 16$

for  $\gamma = 8$  dB, two sensors are sufficient to nearly reach this cap. This is different for higher-order modulation schemes. In the case of  $\gamma = 8$  dB, there occurs a gap between the 4-ASK and the 8-ASK or 16-ASK. This can be explained by the fact that  $I(\mathcal{X}; \mathcal{Y})$  is larger for higher-order modulation schemes in this SNR domain. However, the gain in relevant information becomes smaller for higher-order modulation schemes resulting in nearly no gain comparing the 8-ASK with the 16-ASK. Naturally, this could change for larger SNRs or more sensors. Note that the gain is generally limited due to the rate constraints of  $C_{\text{sum}} = 4.0$  bit/s/Hz. For a signal-to-noise ratio of  $\gamma_m = 3$  dB, there is no difference between the 4-ASK, 8-ASK, and the 16-ASK. This is caused by the fact that in this SNR domain, the 4-ASK, 8-ASK, and 16-ASK contain nearly the same information  $I(\mathcal{X}; \mathcal{Y})$ . Summarizing, it can be stated that the GDIB algorithm can be applied for arbitrary discrete probability distributions. Since all considered input distributions deliver qualitatively the same results, it is sufficient to restrict on lower-order relevant signals, which results in lower memory requirements during simulations. This is why most of the following results are only obtained for a 4-ASK relevant signal.

#### 4.2.7 Influence of Optimization Order for Asymmetric Scenarios

The influence of the Wyner-Ziv coding strategy needs to be investigated. Therefore, the performance of the GDIB algorithm using different optimization orders to optimize the sensors is analyzed. Naturally, for this investigation, it is not meaningful to consider symmetric scenarios, as they would provide the same solution for each optimization order. Hence, two different asymmetric scenarios with  $M = 4$  sensors are examined. The first scenario combines sensors with bad SNRs  $\gamma_m$  with low link capacities  $C_m$ . Therefore, good sensors also have good forward channel conditions, and bad sensors also have bad forward channel conditions. The second scenario, however, combines sensors with bad SNRs with high link capacities. Therefore, good sensors have very poor forward channel conditions, while bad sensors have very good forward channel conditions.

Figure 4.12 illustrates the performance of the GDIB algorithm, i.e., the relevant information after GDIB optimization for  $M = 4$  sensors and all permutations of the optimization order. The blue bars represent the first scenario, while the red bars represent the second scenario. Considering the first scenario, it seems that the Wyner-Ziv coding strategy, i.e., the optimization order, does not have a big impact on the performance of the GDIB algorithm. However, the second scenario shows that the different optimization orders can highly influence the performance of the GDIB algorithm in asymmetric scenarios. In general, it can be observed that the performance of the second scenario is worse than the performance of the first one. This can be expected since, in the second scenario, accurate measurements have to be strongly compressed, while unreliable measurements can only contribute little to the overall performance, even if the corresponding link capacities are high. Considering the different permutations within the second scenario, no clear conclusion about the optimal Wyner-Ziv coding strategy can be drawn. However, a good solution can be expected when starting the optimization with the best forward channel conditions, i.e., the lowest compression.



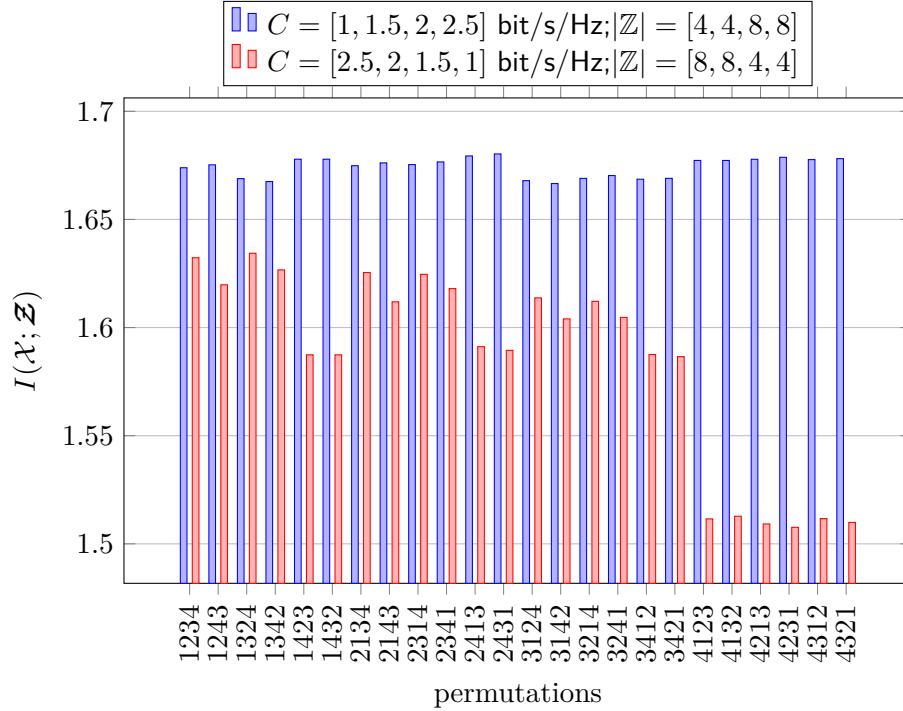


Figure 4.12: Relevant mutual information for an asymmetric scenario with  $M = 4$  sensors, SNRs  $\gamma_m = [2, 4, 6, 8]$  dB and  $|\mathbb{X}| = 4$  using GDIB optimization

#### 4.2.8 Robustness of the GDIB Algorithm

So far, all simulations have modeled the additive measurement noise as white and Gaussian distributed. This is a natural choice in communications for channel noise due to the Central Limit Theorem. For measurement noise, this might not be an obvious decision. However, white and Gaussian distributed noise minimizes the capacity of a point-to-point additive noise channel, i.e., it minimizes the mutual information  $I(\mathcal{X}; \mathcal{Y}_m)$ . This means that choosing white and Gaussian distributed noise represents the worst case also for measurement noise.

Regarding the robustness of the proposed optimization algorithms, the measurement noise is a crucial factor since it directly influences the joint probability  $p(x, y_m) = p(x)p(y_m|x)$ . This subsection investigates the influence of the signal-to-noise ratio  $\gamma_m$  on the optimization approach. In particular, how robust the optimization reacts on wrong assumptions regarding the noise power  $\sigma_{w_m}^2$ . This can either be caused by not knowing the signal-to-noise ratio perfectly in advance or by changing over time.

As the GDIB approach maximizes the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  while still maintaining a desired compression rate  $I(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{< m}) \leq C_m$  the question arises, how a measurement SNR mismatch influences this behavior. Therefore, similar to the scalar case in Section 3.4, the optimization SNR does not match the actual application SNR. Given a fixed application measurement signal-to-noise ratio  $\gamma_m^{\text{app}}$  with a corresponding channel

likelihood  $p(y_m|x)$ , a quantizer  $\tilde{p}(z_m|y_m)$  optimized for a different  $\gamma_{\text{opt}}$  with  $\tilde{p}(y_m|x)$  is applied. The resulting relevant mutual information is given as

$$\tilde{I}(\mathcal{X}; \mathcal{Z}) = \mathbb{E}_{\mathcal{X}, \mathcal{Z}} \left[ \log_2 \frac{\tilde{p}(\mathbf{z}|x)}{\tilde{p}(\mathbf{z})} \right] \quad (4.18)$$

with

$$\tilde{p}(\mathbf{z}|x) = \prod_{m=1}^M \tilde{p}(z_m|x) \quad (4.19)$$

with

$$\tilde{p}(z_m|x) = \sum_{y_m} \tilde{p}(z_m|y_m)p(y_m|x) \quad (4.20)$$

and

$$\tilde{p}(\mathbf{z}) = \sum_x \tilde{p}(\mathbf{z}|x)p(x) . \quad (4.21)$$

The resulting compression rate for a specific sensor  $m$  can be calculated by

$$\tilde{I}(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{<m}) = \mathbb{E}_{\mathcal{Y}, \mathcal{Z}_{\leq m}} \left[ \log_2 \frac{\tilde{p}(z_m|y_m)}{\tilde{p}(z_m|\mathbf{z}_{<m})} \right] \quad (4.22)$$

with

$$\tilde{p}(\mathbf{z}_{\leq m}) = \sum_x p(x) \prod_{i=1}^m \tilde{p}(z_i|x) \quad (4.23)$$

and

$$\tilde{p}(z_m|\mathbf{z}_{<m}) = \frac{\tilde{p}(\mathbf{z}_{\leq m})}{\sum_{z_m} \tilde{p}(\mathbf{z}_{\leq m})} . \quad (4.24)$$

The left diagram of Figure 4.13 illustrates the relevant mutual information  $\tilde{I}(\mathcal{X}; \mathcal{Z})$  versus the optimization signal-to-noise ratio  $\gamma_m^{\text{opt}}$  for different application SNRs  $\gamma_m^{\text{app}}$ . The simulation is done for a network of  $M = 5$  sensors and a 4-ASK relevant signal. The sum-rate is chosen to be  $C_{\text{sum}} = 2.5$  bit/s/Hz which is equally distributed on each forward link  $C_m = \frac{C_{\text{sum}}}{M} = 0.5$  bit/s/Hz. Therefore, each sensor needs to perform a strong compression to match the rate constraints. The other cardinalities are defined as  $|\mathcal{Y}_m| = 512$  and  $|\mathcal{Z}_m| = 4$ . Naturally, a larger application SNR  $\gamma_m^{\text{app}}$  results in an overall better performance with larger relevant mutual information  $\tilde{I}(\mathcal{X}; \mathcal{Z})$ . Similar to the scalar case, the general expectation on these curves is that the maximum of a curve for a specific  $\gamma_m^{\text{opt}}$  should lie on the position of  $\gamma_m^{\text{opt}} = \gamma_m^{\text{app}}$ . However, this behavior can not be observed in Figure 4.13. It seems that larger relevant mutual information correlates with a larger optimization SNR independent of the application SNR. To understand this behavior, the compression rates depicted in the right diagrams of Figure 4.13 have to be analyzed. Here, each diagram

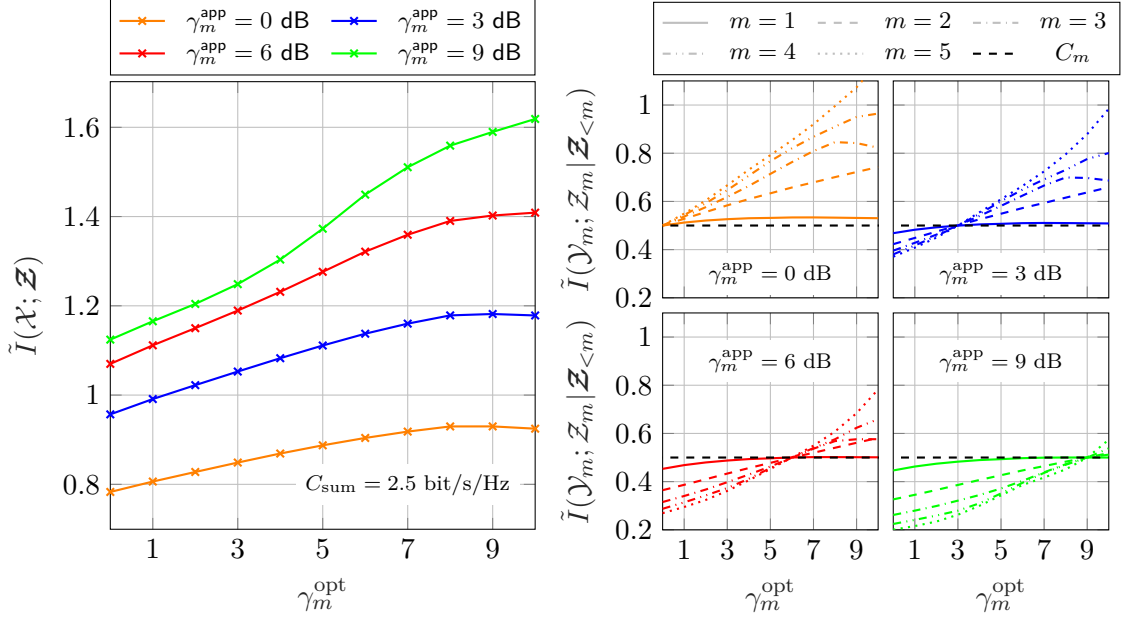


Figure 4.13: Measurement channel mismatch applying the GDIB algorithm for  $M = 5$  sensors with  $C_{\text{sum}} = 2.5$  bit/s/Hz equally distributed on each forward link  $C_m = \frac{C_{\text{sum}}}{M}$ ,  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}_m| = 512$  and  $|\mathbb{Z}_m| = 4$

represents the compression rates  $\tilde{I}(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{<m})$  versus the optimization SNR  $\gamma_m^{\text{opt}}$  of all sensors  $m = 1, \dots, m = 5$  for a specific application SNR  $\gamma_m^{\text{app}}$ . The black dashed line shows the capacity bound  $C_m$ , which defines the maximum allowed compression rate. Naturally, for  $\gamma_m^{\text{app}} = \gamma_m^{\text{opt}}$  the rate constraints are perfectly matched, i.e.,  $\tilde{I}(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{<m}) \approx C_m$ . However, if  $\gamma_m^{\text{opt}} \neq \gamma_m^{\text{app}}$  this does not hold anymore. In particular, if  $\gamma_m^{\text{opt}} < \gamma_m^{\text{app}}$  the compression rate falls below the rate constraint  $\tilde{I}(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{<m}) < C_m$  while for  $\gamma_m^{\text{opt}} > \gamma_m^{\text{app}}$  the compression rate exceeds the rate constraint  $\tilde{I}(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{<m}) > C_m$ . In fact, for a low optimization SNR  $\gamma_m^{\text{opt}}$  the measurement channel likelihood  $\tilde{p}(y_m|x)$  is assumed to be broad while for a good  $\gamma_m^{\text{opt}}$  it is assumed to be slim. If the optimization SNR  $\gamma_m^{\text{opt}}$  is better than the application SNR  $\gamma_m^{\text{app}}$  the true channel likelihood  $p(y_m|x)$  is broader than  $\tilde{p}(y_m|x)$  leading to an increase in compression rate  $\tilde{I}(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{<m})$ . The observed effect becomes more significant for late sensors in the optimization chain. This behavior explains why the maximum of the relevant mutual information  $\tilde{I}(\mathcal{X}; \mathcal{Z})$  in the left diagram of Figure 4.13 is shifted to the right, especially for lower application SNRs.

Figure 4.14 illustrates results for the same simulation, but for a sum-rate of  $C_{\text{sum}} = 5$  bit/s/Hz. Therefore, the individual rate constraints increase to  $C_m = 1$  bit/s/Hz resulting in a weaker compression. Considering the left diagram, it becomes obvious that the shifted maxima observed in Figure 4.13 are still noticeable. However, this effect is not as pronounced as before. Considering the right diagram, again, it can be observed that if  $\gamma_m^{\text{opt}} > \gamma_m^{\text{app}}$  the compression rate of each sensor exceeds the individual capacity constraint  $\tilde{I}(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{<m}) > C_m$ .

Figure 4.15 illustrates results for the same simulation, but for a sum-rate of  $C_{\text{sum}} = 7.5$  bit/s/Hz, leading to individual rate constraints of  $C_m = 1.5$  bit/s/Hz. The left diagram reveals that in this case, the measurement SNR mismatch has a much lower impact on

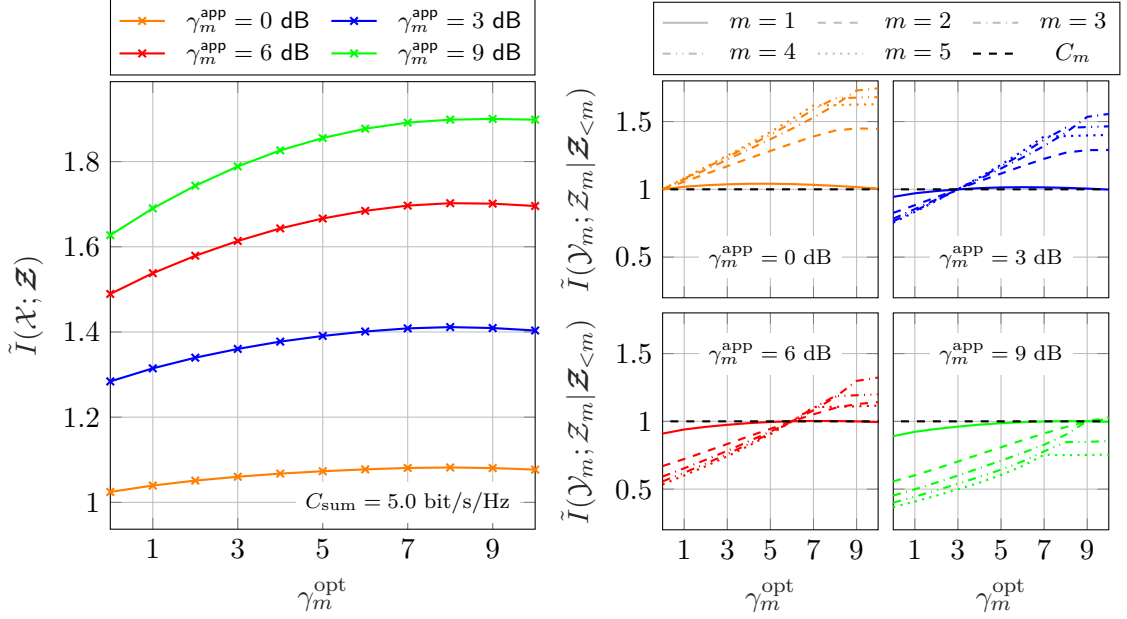


Figure 4.14: Measurement channel mismatch applying the GDIB algorithm for  $M = 5$  sensors with  $C_{\text{sum}} = 5.0$  bit/s/Hz equally distributed on each forward link  $C_m = \frac{C_{\text{sum}}}{M}$ ,  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}_m| = 512$  and  $|\mathbb{Z}_m| = 4$

the relevant mutual information  $\tilde{I}(\mathcal{X}; \mathcal{Z})$  than for lower sum-rates. It can be observed that  $\tilde{I}(\mathcal{X}; \mathcal{Z})$  is constant in a wide range, especially for lower application SNRs. For larger application SNRs, this range becomes a bit smaller. However, considering the right diagram reveals that for lower application SNRs like  $\gamma_m^{\text{app}} = \{0, 3\}$  dB, the compression rates are not fulfilled.

Finally, Figure 4.16 illustrates results for a sum-rate of  $C_{\text{sum}} = 10.0$  bit/s/Hz, i.e.,  $C_m = 2.0$  bit/s/Hz. Since the relevant signal is modeled as a 4-ASK signal, these rate constraints are very loose restrictions. Similar to Figure 4.15 with a sum-rate of  $C_{\text{sum}} = 7.5$  bit/s/Hz the measurement signal-to-noise ratio mismatch has a very low impact on the relevant mutual information  $\tilde{I}(\mathcal{X}; \mathcal{Z})$ . In contrast to lower sum-rates, for  $C_{\text{sum}} = 10.0$  bit/s/Hz, the rate constraints are not violated anymore.

Summarizing, it can be stated that in the considered scenario the measurement SNR mismatch has a higher impact on stricter rate constraints. In these cases, the individual rate constraints are violated if the optimization SNR is larger than the application SNR  $\gamma_m^{\text{opt}} > \gamma_m^{\text{app}}$ . In order to cope with this behavior, two possible strategies can be pursued, similar to the scalar case in Section 3.4. First, a compression back-off could be considered during the optimization, ensuring that the rate constraint is fulfilled in any case. Therefore, the optimization needs to be performed for a lower rate constraint than the actual  $C_m$ . The drawback of this strategy is that the required back-off is hard to determine in advance. The second approach is more straightforward as it suggests an optimization for a lower SNR. In this case, it is very likely that even if the measurement channel has a significant mismatch, the compression rate is still lower than the rate constraint  $C_m$ . However, both approaches result in a loss of relevant mutual information.

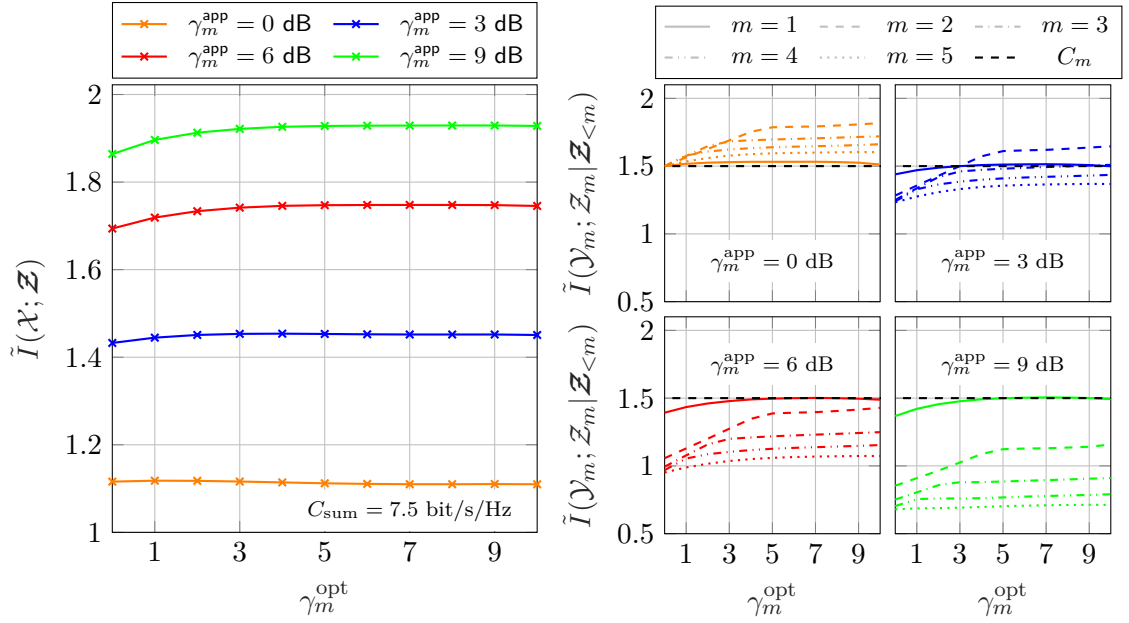


Figure 4.15: Measurement channel mismatch applying the GDIB algorithm for  $M = 5$  sensors with  $C_{\text{sum}} = 7.5$  bit/s/Hz equally distributed on each forward link  $C_m = \frac{C_{\text{sum}}}{M}$ ,  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}_m| = 512$  and  $|\mathbb{Z}_m| = 4$

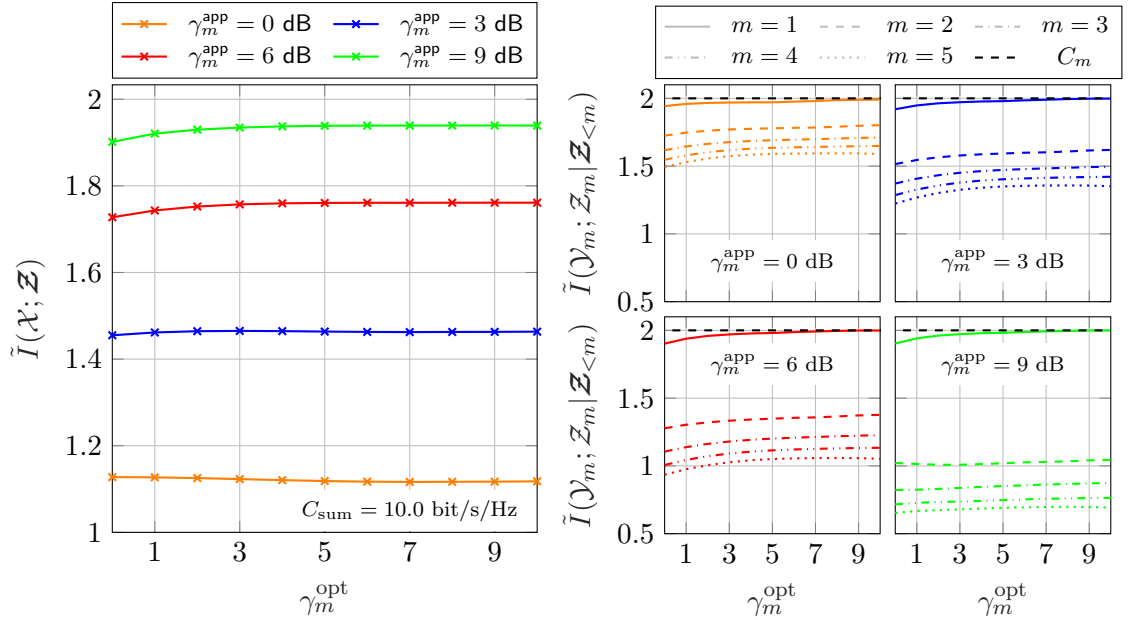


Figure 4.16: Measurement channel mismatch applying the GDIB algorithm for  $M = 5$  sensors with  $C_{\text{sum}} = 10.0$  bit/s/Hz equally distributed on each forward link  $C_m = \frac{C_{\text{sum}}}{M}$ ,  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}_m| = 512$  and  $|\mathbb{Z}_m| = 4$

### 4.2.9 Greedy Distributed Information Bottleneck Algorithm for Deterministic Mappings

The original GDIB algorithm produces stochastic mappings  $p(z_m|y_m)$ . However, stochastic quantization is challenging from an implementation point of view. Therefore, it might be preferable to design deterministic quantizers instead. Following the derivation for the scalar deterministic iterative IB approach in Section 3.4, a similar approach can be pursued for the distributed case. This means with the choice of  $\beta_m = 0$ , the utility functions in (4.15a) - (4.15b) focus only on the preservation of relevant information. Moreover, the second term in the statistical distance (4.17) can be neglected, and the sum in the denominator of the update equation (4.16) is dominated by the smallest KL divergence. Hence, for each sample,  $y_m$ , the conditional probabilities  $p(z_m|y_m)$  tend to zero for all  $z_m$  except for the one with the smallest KL divergence, which tends to be one. With

$$z_m^* = \arg \min_{z_m} \mathbb{E}_{\mathcal{Z}_{<m}|y_m} \left[ D_{\text{KL}} [p(x|y_m, \mathbf{z}_{<m}) || p(x|\mathbf{z}_{<m})] \right], \quad (4.25)$$

the update equation becomes

$$p(z_m|y_m) = \begin{cases} 1 & \text{for } z_m = z_m^* \\ 0 & \text{else} \end{cases} \quad (4.26)$$

and provides a deterministic mapping. Since the original GDIB algorithm adjusts the compression rates by varying  $\beta_m$ , in the deterministic case, the compression needs to be adjusted by choosing an appropriate cardinality  $|\mathcal{Z}_m|$ . Within this thesis, this variation of the GDIB approach is named *deterministic GDIB*. In Section 3.4, it was shown that the scalar deterministic iterative IB approach produces equally distributed output clusters. Naturally, this is the same for the deterministic GDIB approach when optimizing the first sensor. However, for later sensors, this might not necessarily hold anymore since the optimization exploits the mappings of previously designed quantizers. Therefore,  $|\mathcal{Z}_m|$  does not directly determine the compression rate to  $\log_2 |\mathcal{Z}_m|$  for  $m > 1$  but to  $H(\mathcal{Z}_m)$ . The adapted flowchart for the deterministic GDIB approach is given in Figure 4.17. Basically, the main difference to the original GDIB approach lies in the bisection search, which is performed in the "loop to adjust  $|\mathcal{Z}_m|$ ", which now adjusts the cardinality of  $\mathcal{Z}_m$  of the specific sensor. The extended Blahut-Arimoto algorithm including (4.25) and (4.26) is given in Algorithm 5.

**Performance Analysis Considering the Sum-Rate Only:** Figure 4.18 illustrates the relevance compression curve, i.e., the vector compression rate  $I(\mathcal{Y}; \mathcal{Z})$  versus the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  for the original GDIB and the deterministic GDIB solution. Here, a symmetric scenario with  $M = 3$  sensors, a relevant 4-ASK signal, and measurement SNRs  $\gamma_m = 8$  dB is considered. The red curve represents the original GDIB solution, where the output cardinality  $|\mathcal{Z}_m| = 16$  is fixed for all sensors, and the compression rate is adjusted by varying  $\beta_m$ . The blue marks represent the deterministic case,

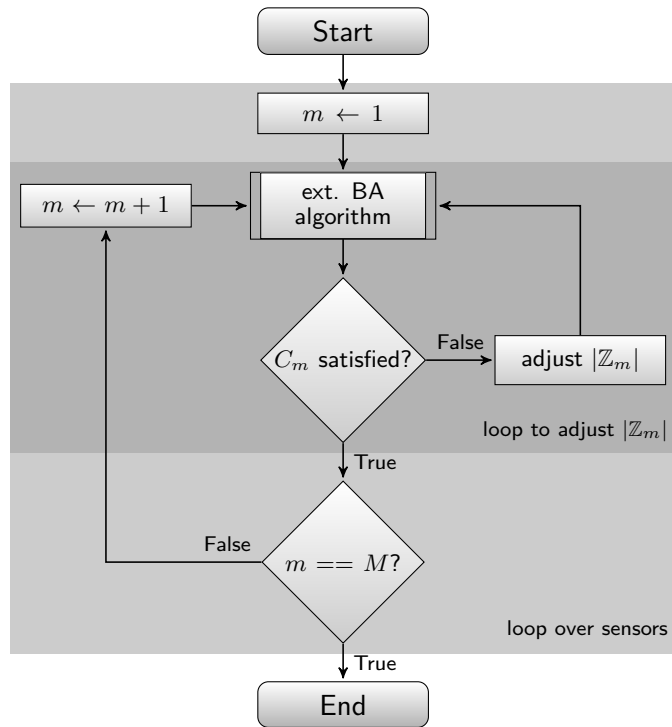


Figure 4.17: Flowchart of the deterministic GDIB algorithm

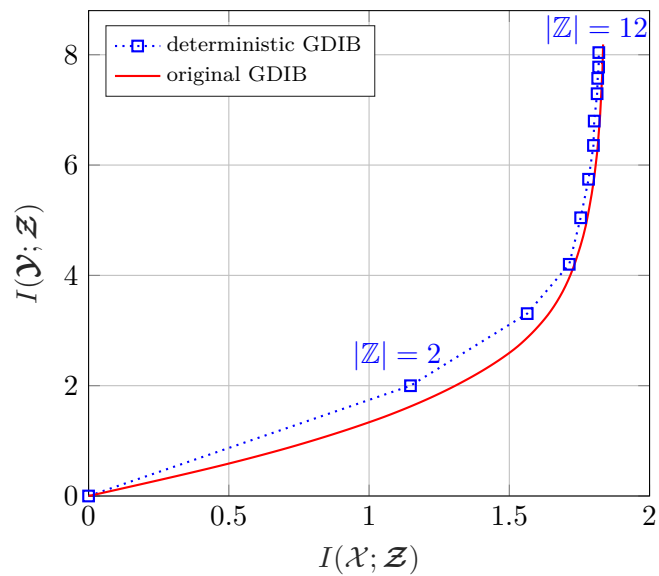


Figure 4.18: Compression-Relevance curve for  $M = 3$ ,  $|\mathcal{X}| = 4$ ,  $|\mathcal{Y}_m| = 64$  and  $\gamma_m = 8$  dB; for stochastic GDIB:  $|\mathcal{Z}_m| = 16$

---

**Algorithm 5:** Extended Blahut-Arimoto algorithm for the deterministic GDIB algorithm

---

```

input      :  $m, p(y_m, x), p^{\text{init}}(z_m|y_m), p(\mathbf{z}_{<m}|x), \epsilon$ 
output    :  $p(z_m|y_m) \in \{0, 1\}$ 
1 begin
   initialization:
        $p(z_m|y_m)^{(0)} \leftarrow p^{\text{init}}(z_m|y_m),$ 
        $l \leftarrow 1$ 
2    $p(\mathbf{z}_{<m}, y_m, x) = p(\mathbf{z}_{<m}|x)p(y_m, x)$ 
3    $p(x|\mathbf{z}_{<m}, y_m) = p(\mathbf{z}_{<m}, y_m, x) / \sum_x p(\mathbf{z}_{<m}, y_m, x)$ 
4    $p(\mathbf{z}_{<m}|y_m) = \sum_x p(\mathbf{z}_{<m}, y_m, x) / \sum_x p(y_m, x)$ 
5   do
6       // calculate  $p(\mathbf{z}_{\leq m}, y_m, x)$ 
7        $p(\mathbf{z}_{\leq m}, y_m, x)^{(l)} = p(\mathbf{z}_{<m}, y_m, x)p(z_m|y_m)^{(l-1)}$ 
8       // KL divergence  $D_{\text{KL}}(y_m, \mathbf{z}_{\leq m})$  of (4.17)
9        $p(x|\mathbf{z}_{\leq m})^{(l)} = \sum_{y_m} p(\mathbf{z}_{\leq m}, y_m, x)^{(l)} / \sum_{y_m, x} p(\mathbf{z}_{\leq m}, y_m, x)^{(l)}$ 
10       $D_{\text{KL}}(y_m, \mathbf{z}_{\leq m})^{(l)} = \sum_x p(x|\mathbf{z}_{<m}, y_m) \cdot \log \frac{p(x|\mathbf{z}_{<m}, y_m)}{p(x|\mathbf{z}_{\leq m})^{(l)}}$ 
11      // find minimum of  $D_{\text{KL}}(y_m, \mathbf{z}_{\leq m})$  for all samples  $y_{m_i} \in \mathbb{Y}_m$ 
12      for  $y_{m_i} \in \mathbb{Y}_m$  do
13           $z_m^*(y_{m_i})^{(l)} = \arg \min_{z_m} \sum_{\mathbf{z}_{<m}} p(\mathbf{z}_{<m}|y_m) D_{\text{KL}}(y_m, \mathbf{z}_{\leq m})^{(l)}$ 
14          // update quantizer for specific  $y_{m_i}$ 
15           $p(z_m^*(y_{m_i})^{(l)}|y_{m_i})^{(l)} = 1$ 
16       $l \leftarrow l + 1$ 
17  while  $D_{\text{JS}}[p^{(l)}(z_m|y_m) || p^{(l-1)}(z_m|y_m)] > \epsilon$ 

```

---

where  $\beta_m = 0$  for all sensors, and the compression rate is adjusted by varying the cardinality of  $\mathbb{Z}_m$ . In particular,  $|\mathbb{Z}_m|$  is linearly increased from  $|\mathbb{Z}_m| = 1$  to  $|\mathbb{Z}_m| = 12$ . To simplify the analysis, just the sum-rate is considered. Hence,  $\beta_m = \beta$  and  $|\mathbb{Z}_m| = |\mathbb{Z}|$  hold for all sensors. In this way, the overall performance is not influenced by individual rate constraints. For this simulation, no bisection search is needed. Therefore, the individual compression rates of each sensor might differ, comparing the stochastic with the deterministic case. Figure 4.18 reveals that the solution of the deterministic GDIB solution is very close to the stochastic one, especially for larger compression rates. The difference confirms the result for the scalar case in Section 3.4 as well as the statement in [Slo02] that stochastic mappings usually perform better than deterministic mappings since the global optimum is stochastic in general.

**Performance for Individual Link Capacities:** Figure 4.19 illustrates a comparison of the original GDIB resulting in stochastic mappings and the deterministic GDIB for a setup with individual link capacities  $C_m$ . Therefore, a symmetric scenario of  $M = 5$  sensors,  $\gamma_m = 6$  dB, and link capacities of  $C_m = \frac{C_{\text{sum}}}{M}$  has been analyzed. The relevant information is chosen to be a 4-ASK signal. The left diagram of Figure 4.19 shows the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  versus different sum-rates  $C_{\text{sum}}$ . In red, the performance of the



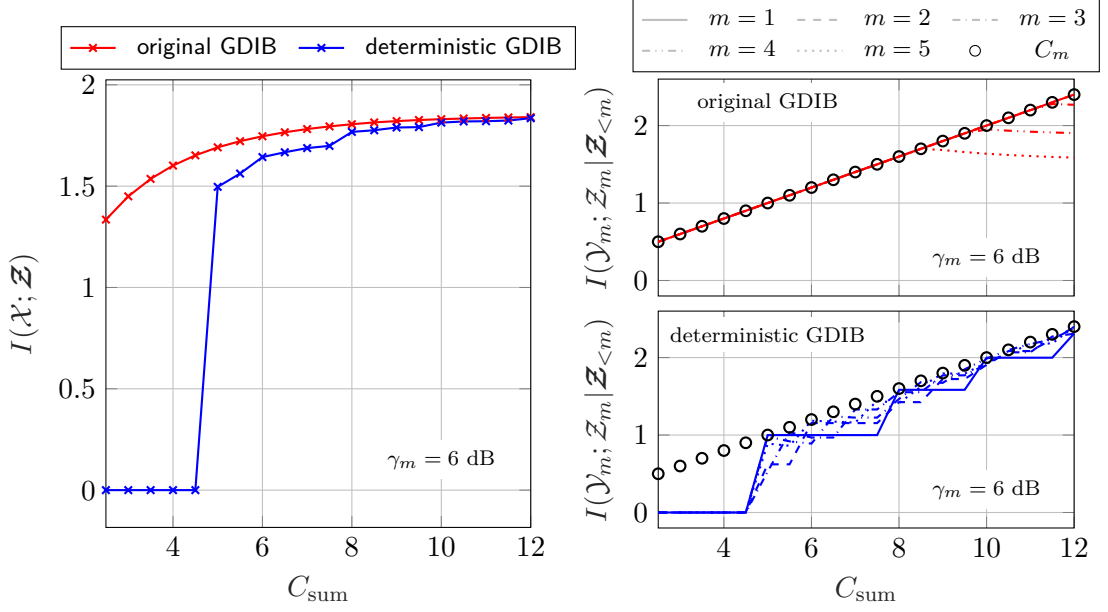


Figure 4.19: Comparison of original GDIB and deterministic GDIB with individual link capacities  $C_m = \frac{C_{\text{sum}}}{M}$  for different sum-rates;  $M = 5$ ,  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}_m| = 64$  and  $\gamma_m = 6$  dB; for stochastic GDIB:  $|\mathbb{Z}_m| = 16$

original GDIB is shown, while the performance of the deterministic approach is depicted in blue. The individual compression rates for the different sum-rates are adjusted using the bisection search. In the case of the original GDIB,  $\beta_m$  is varied for a fixed cardinality  $|\mathbb{Z}_m| = 16$ . For the deterministic case,  $\beta_m = 0$  holds and the compression rate is adjusted by changing  $|\mathbb{Z}_m|$ . Note that only integer values can be used to adjust the cardinality, which may result in compression rates falling below the individual link capacity  $C_m$ . It becomes obvious that the performance of the original GDIB approach outperforms the deterministic GDIB approach, especially for lower sum-rates. This could be expected since the performance of stochastic mappings is better than that of deterministic mappings. Having a closer look at the deterministic GDIB approach, it can be observed that for  $C_{\text{sum}} < 5$  bit/s/Hz, the deterministic approach fails completely. Since the sum capacity is equally distributed on the different links, the individual link capacity is  $C_m < 1$  bit/s/Hz in this area. As already pointed out in the scalar scenario in Section 3.4, the first sensor produces mappings with equally distributed output clusters. Since two output clusters represent the lowest possible compression rate for  $I(\mathcal{Y}_1; \mathcal{Z}_1) > 0$  bit/s/Hz the minimum achievable compression rate is  $I(\mathcal{Y}_1; \mathcal{Z}_1) = H(\mathcal{Z}_1) = 1$  bit/s/Hz. Everything below causes the algorithm to choose just one output cluster, leading to zero relevant mutual information  $I(\mathcal{Y}_1; \mathcal{Z}_1) = 0$ . Moreover, if the first sensor cannot preserve any information, the second sensor would treat the first one as not existing. Hence the second sensor experiences the same problem leading to a complete network failure.

The right diagram in Figure 4.19 illustrates compression rates  $I(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{<m})$  for each sensor in the network for the same simulation. The top diagram represents the original GDIB approach, while the bottom diagram represents the deterministic GDIB approach. The round markers show the individual link capacity for each sensor  $C_m$ . Comparing

the two approaches reveals the difference in performance. The original GDIB approach always hits the target compression rate with all sensors. Only for large sum-rates the last sensors in the optimization chain cannot provide enough new information to exhaust the link capacity  $C_m$ . All information about the relevant signal  $x$  has already been sent by previous sensors. Naturally, this might change for different relevant signals or SNRs. On the contrary, the deterministic approach does not exactly hit the compression rate since the cardinality  $|\mathbb{Z}_m|$  can only be adjusted in integer steps. If the first sensor has a large gap to the capacity  $C_1$ , subsequent sensors can perform better. It seems as if they try to compensate the loss of the first sensor. If, however, the first sensor almost hits the capacity  $C_1$ , later sensors have larger gaps to the target rates. Moreover, this is directly linked to the relevant information as well. This gap to the capacity  $C_m$  leads to the performance loss in relevant mutual information  $I(\mathcal{X}; \mathbf{Z})$  shown in the left diagram in Figure 4.19. Moreover, the solid blue curve in the right diagram confirms the fact that the first sensor produces equally distributed output clusters.

### 4.3 Reduced-Memory Complexity GDIB Algorithm

Basically, the previously defined GDIB algorithm can be applied for distributed setups of any network size. However, as the algorithm exploits the mappings of previously designed quantizers, the dimensionality of the pmfs in the statistical distance measure  $d_{\beta_m}(y_m, z_m)$  in (4.17) used to update the quantizer mappings in (4.16) are dependent on the network size. More precisely, by increasing the network size, the dimensionality of the involved vector  $\mathbf{Z}_{<m}$  increases. Therefore, a linear increase of the network size leads to an exponential increase in the number of elements in the representing data structure of the involved pmfs. To give a short example, considering a network of  $M = 10$  sensors with cardinalities  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}_m| = 64$  and  $|\mathbb{Z}_m| = 8 \forall m$ , when optimizing the last sensor the tuple  $(y_M, \mathbf{z}_{<M})$  already comprise  $64 \cdot 8^9 \approx 8.59 \cdot 10^9$  elements. In order to store the pmf  $p(x|y_M, \mathbf{z}_{<M})$  with 8 byte for double precision, it requires 256 GiB<sup>4</sup>. The given example emphasizes that the GDIB algorithm will definitely run into memory problems for larger network sizes.

In order to ensure feasibility even for larger networks, an approach to compress the high dimensional  $\mathbf{Z}_{<m}$  onto a single dimensional  $\mathbf{Z}_{<m}^*$  of appropriate cardinality  $|\mathbb{Z}_{<m}^*|$  is introduced in [Ste+21b; Ste+21a]. Since  $\mathbf{Z}_{<m}^*$  shall be as informative as possible about the relevant signal  $\mathcal{X}$ , the IB method can be applied for compression. However, since there are no restrictions on the compression rate, the trade-off parameter can be chosen as  $\beta = 0$  in order to focus only on the preservation of relevant information. The optimization problem for compressing  $\mathbf{Z}_{<m}$  can be formulated as

$$L_C[p(z^*|\mathbf{z})] = \max_{p(z^*|\mathbf{z})} I(\mathcal{X}; \mathbf{Z}^*). \quad (4.27)$$

<sup>4</sup>1 GiB = 1024 MiB, 1 MiB = 1024 KiB, 1 KiB = 1024 byte

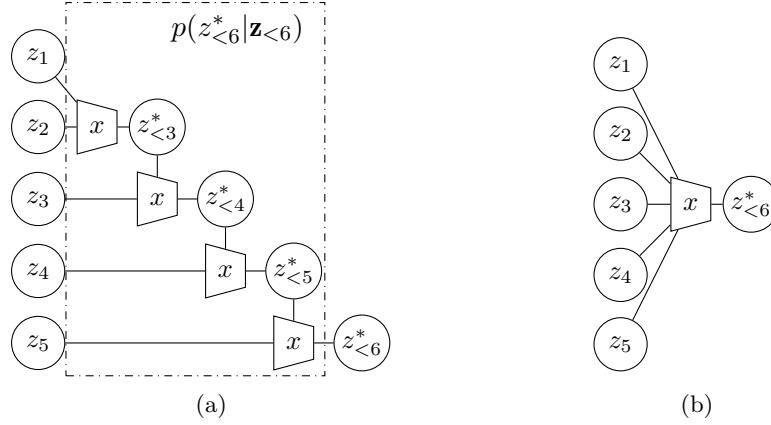


Figure 4.20: (a) sequential compression of  $\mathbf{z}_{<6}$  for sensor 6 (b) one-step compression

The resulting compressed  $\mathcal{Z}_{<m}^*$  shall be as informative about the relevant signal  $\mathcal{X}$  as the multidimensional  $\mathcal{Z}_{<m}$ . Therefore, they can be exchanged in the definition of the statistical distance  $d_{\beta_m}(y_m, z_m)$  in (4.17), leading to a compressed version

$$d_{\beta_m}^c(y_m, z_m) := \mathbb{E}_{\mathcal{Z}_{<m}^* | y_m} \left[ \frac{1}{\beta_m} \cdot D_{\text{KL}} [p(x | y_m, z_{<m}^*) || p(x | z_m, z_{<m}^*)] - \log p(z_m | z_{<m}^*) \right]. \quad (4.28)$$

It becomes obvious that in this modified definition of the statistical distance  $d_{\beta_m}^c(y_m, z_m)$  no multivariate vector  $\mathcal{Z}_{<m}$  occurs anymore. Therefore, storing these pmfs in appropriate data structures requires much less memory than in (4.17).

Using this modified definition of the statistical distance in (4.28) a reduced-memory complexity version of the extended Blahut-Arimoto algorithm to optimize a specific quantizer  $m$  for a specific  $\beta_m$  is given in Algorithm 6. The compression of  $\mathcal{Z}_{<m}$  can be performed in advance of this algorithm. This compression delivers the input pmf  $p(z_{<m}^*, x)$ . Two different compression schemes will be discussed in the next subsections. The KL divergence in (4.28) is determined in lines 8 to 10 by means of the joint pmf  $p(z_{<m}^*, z_m, y_m, x)$  calculated in lines 6 and 7. The statistical distance  $d_{\beta_m}^c(z_m, y_m)$  computed in lines 11 to 13 is used to update the quantizer mapping  $p(z_m | y_m)$ . As in the original GDIB algorithm, the reduced-memory complexity version of the Blahut-Arimoto-like algorithm in Algorithm 6 stops when a certain stopping criterion is fulfilled. Applying this reduced-memory complexity version of the Blahut-Arimoto algorithm in the GDIB algorithm of Figure 4.1 defines the reduced-memory complexity GDIB algorithm.

### 4.3.1 Sequential Compression Scheme

As already mentioned, the compression of  $\mathbf{z}_{<m}$  to  $z_{<m}^*$  is performed using the IB method. In particular, to overcome the curse of dimensionality, a *sequential compression* scheme can be performed. Here,  $\mathbf{z}_{<m}$  is compressed in a sequential way using a tree-based structure. To achieve this, the compression is always performed for two variables at a time. A graphical tool to illustrate this are IB graphs [LSB16a]. Here, the trapezoid represents

**Algorithm 6:** Extended Blahut-Arimoto Algorithm with Reduced-Memory Complexity

---

```

input      :  $m, p(y_m, x), p^{\text{init}}(z_m|y_m), \beta_m, \epsilon,$ 
                $p(z_{<m}^*, x)$ 
output    :  $p(z_m|y_m) \in [0, 1]$ 
1 begin
   initialization:
                $p(z_m|y_m)^{(0)} \leftarrow p^{\text{init}}(z_m|y_m),$ 
                $l \leftarrow 1$ 
2    $p(z_{<m}^*, y_m, x) = p(z_{<m}^*, x)p(y_m, x)/p(x)$ 
3    $p(x|z_{<m}^*, y_m) = p(z_{<m}^*, y_m, x) / \sum_x p(z_{<m}^*, y_m, x)$ 
4    $p(z_{<m}^*|y_m) = \sum_x p(z_{<m}^*, y_m, x) / p(y_m)$ 
5   do
6     // calculate  $p(z_{<m}^*, z_m, y_m, x)$ 
7      $p(z_{<m}^*, z_m, y_m, x)^{(l)} = p(z_{<m}^*, y_m, x)p(z_m|y_m)^{(l-1)}$ 
8     // KL divergence  $D_{\text{KL}}(y_m, z_m, z_{<m}^*)$  of (4.28)
9      $p(x|z_{<m}^*, z_m)^{(l)} = \sum_{y_m} p(z_{<m}^*, z_m, y_m, x)^{(l)} / \sum_{y_m, x} p(z_{<m}^*, z_m, y_m, x)^{(l)}$ 
10     $D_{\text{KL}}(y_m, z_m, z_{<m}^*)^{(l)} = \sum_x p(x|z_{<m}^*, z_m) \cdot \log \frac{p(x|z_{<m}^*, y_m)}{p(x|z_{<m}^*, z_m)^{(l)}}$ 
11    // distance  $d_{\beta_m}^c(z_m, y_m)$  (4.28)
12     $p(z_m|z_{<m}^*)^{(l)} = \sum_{y_m, x} p(z_{<m}^*, z_m, y_m, x)^{(l)} / \sum_{y_m, x} p(z_{<m}^*, y_m, x)$ 
13     $d_{\beta_m}^c(z_m, y_m)^{(l)} =$ 
        
$$\sum_{z_{<m}^*} p(z_{<m}^*|y_m) \cdot \left[ \frac{1}{\beta_m} D_{\text{KL}}(y_m, z_m, z_{<m}^*)^{(l)} - \log p(z_m|z_{<m}^*)^{(l)} \right]$$

14    // update quantizer  $p(z_m|y_m)$ 
15     $p(z_m|y_m)^{(l)} = \frac{1}{\sum_z e^{-d_{\beta_m}^c(z_m, y_m)^{(l)}}} e^{-d_{\beta_m}^c(z_m, y_m)^{(l)}}$ 
16     $l \leftarrow l + 1$ 
17 while  $D_{\text{JS}}[p^{(l)}(z_m|y_m) || p^{(l-1)}(z_m|y_m)] > \epsilon$ 

```

---

the IB compression. The relevant random variable is written inside the trapezoid. Since IB graphs are closely related to factor graphs, they denote the involved random variables as variable nodes expressed by circles.

An example is given in Figure 4.20a considering a network of 6 sensors. When optimizing the last sensor, the indexes  $z_1, \dots, z_5$  shall be compressed to  $z_{<6}^*$ . The compression starts with  $z_1$  and  $z_2$ , compressing them to  $z_{<3}^*$ . The corresponding algorithm for the first compression step is given in Algorithm 7. Its output  $z_{<3}^*$  serves as an input for a subsequent joint IB compression with  $z_3$ . Algorithm 8 represents the compression for all sensors  $3 \leq m \leq M$ . This procedure can be repeated for all remaining variables, leading to the tree-based structure.

### 4.3.2 One-Step Compression Scheme

The *one-step compression* scheme serves as a benchmark for the sequential compression scheme introduced in the previous subsection. Here, the compression is performed in a

---

**Algorithm 7:** Sequential Compression of  $\mathbf{z}_{<3}$  for sensor  $m = 3$ 


---

**input** :  $m, p(z_1|x), p(z_2|x), p(x)$   
**output** :  $p(z_{<3}^*, x)$   
**1 begin**  
**2**  $p(\mathbf{z}_{<3}, x) = p(z_1|x)p(z_2|x)p(x)$   
**3**  $p(z_{<3}^*, x) \leftarrow \text{IB}(p(\mathbf{z}_{<3}, x), \beta = 0)$

---



---

**Algorithm 8:** Sequential Compression of  $\mathbf{z}_{<m}$  for sensor  $m > 3$ 


---

**input** :  $m, p(z_{<m-1}^*|x), p(z_{m-1}|x), p(x)$   
**output** :  $p(z_{<m}^*, x)$   
**1 begin**  
**2**  $p(z_{<m-1}^*, z_{m-1}, x) = p(z_{<m-1}^*|x)p(z_{m-1}|x)p(x)$   
**3**  $p(z_{<m}^*, x) \leftarrow \text{IB}(p(z_{<m-1}^*, z_{m-1}, x), \beta = 0)$

---

single step, mapping  $\mathbf{z}_{<m}$  directly onto  $z_{<m}^*$ . An example is given in Figure 4.20b. The corresponding algorithm for the one-step compression scheme is given in Algorithm 9. First, the joint pmf  $p(\mathbf{z}_{<m}, x)$  has to be determined in line 2, which deals as an input parameter for the IB algorithm, used in line 3 to perform the compression. Note that the joint pmf  $p(\mathbf{z}_{<m}, x)$  still depends on the number of sensors in the network. Therefore, this one-step compression scheme does not solve the curse of dimensionality and becomes infeasible for larger networks.

### 4.3.3 Analysis of Compression Schemes

Figure 4.21 analyzes the evolution of the compressed version  $\mathcal{Z}_{\leq n}^*$  within the sequential optimization scheme of Figure 4.20a. Therefore, it illustrates the mutual information  $I(\mathcal{X}; \mathcal{Z}_{\leq n}^*)$  versus the sequential compression steps  $n$  when optimizing the last sensor in a network of size  $M = 10$ . Here,  $n$  represents a step, i.e., a single run of Algorithm 7 or Algorithm 8 in order to compress  $\mathcal{Z}_{<10}$  to  $\mathcal{Z}_{<10}^*$ . The mutual information  $I(\mathcal{X}; \mathcal{Z}_{\leq n}^*)$  represents the amount of information the compressed version  $\mathcal{Z}_{\leq n}^*$  has about the relevant information  $\mathcal{X}$ . The dotted lines show the case where the  $\mathcal{Z}_{<10}$  is not compressed at all (NC). The solid lines represent the case where the one-step compression scheme (OC) is used for compression. Naturally, these cases do not depend on the compression step  $n$ , which is why they occur as horizontal lines. Finally, the dashed lines represent the case where the sequential compression scheme (SC) is applied. The gap between the uncompressed case and the one-step compression scheme can be interpreted as the general compression loss. Naturally, by increasing the cardinality  $|\mathbb{Z}^*|$ , this loss becomes smaller. In the considered scenario, this loss disappears completely for a cardinality of  $|\mathbb{Z}^*| = 64$ , which is much lower than  $|\mathbb{Z}_{<10}|^9 = 2^{18}$  of the uncompressed case. Using the sequential compression scheme, the amount of information  $\mathbb{Z}^*$  has about  $\mathcal{X}$  increases with each additional step, i.e., with each additional sensor contributing information about  $\mathcal{X}$ . However, if the cardinality  $|\mathbb{Z}^*|$  is too low,  $I(\mathcal{X}; \mathcal{Z}_{\leq n}^*)$  saturates early, and a significant gap between the one-step compression scheme and the sequential compression scheme remains. This gap can be explained

---

**Algorithm 9: One-Step Compression**


---

**input** :  $m, p(z_\nu|x) \forall \nu < m, p(x)$   
**output** :  $p(z_{<m}^*, x)$   
**1 begin**  
**2**  $p(\mathbf{z}_{<m}, x) = p(z_1|x) \cdot \dots \cdot p(z_{m-1}|x)p(x)$   
**3**  $p(z_{<m}^*, x) \leftarrow \text{IB}(p(\mathbf{z}_{<m}, x), \beta = 0)$

---

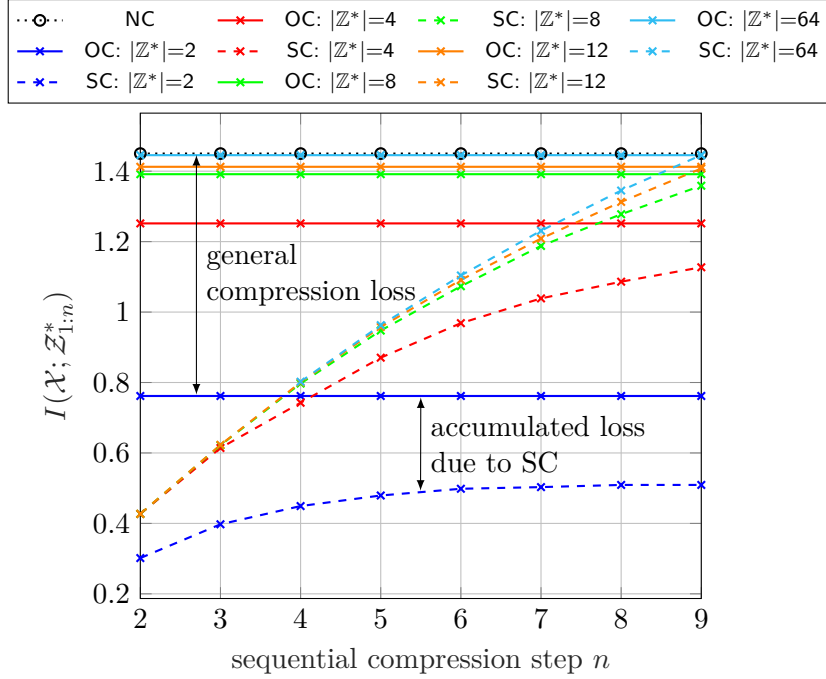


Figure 4.21: Evolution of the compressed version  $\mathcal{Z}_{\leq n}^*$  versus sequential compression steps considering the last sensor; symmetric scenario:  $M = 10$  sensors, sum-rate  $C_{\text{sum}} = 2.5$  bit/s/Hz and  $C_m = \frac{C_{\text{sum}}}{M}$ , SNRs  $\gamma_m = 8$  dB,  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}_m| = 64$  and  $|\mathbb{Z}_m| = 4$ ; Not Compressed (NC), One-Step Compression (OC), Sequential Compression (SC)

by an accumulation of compression losses. By increasing the cardinality  $|\mathbb{Z}^*|$ , this loss decreases, and the sequential compression scheme can perform as good as the one-step compression scheme. Moreover, the memory requirements are reduced significantly when compressing  $\mathcal{Z}_{<10}$  to  $\mathcal{Z}_{<10}^*$ , see Figure 4.23.

#### 4.3.4 Performance of Reduced-Memory Complexity GDIB

The overall performance of the GDIB algorithm when using the described compression schemes is analyzed in Figure 4.22. Here, the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  is depicted versus the number of sensors in the network after optimizing each sensor. In this scenario, the sensors share the channel in an orthogonal way and a round-robin fashion, i.e., the sum-rate  $C_{\text{sum}}$  is equally distributed over all sensors  $C_m = \frac{C_{\text{sum}}}{M}$ . The black dotted line represents the case of the original GDIB algorithm without using the described compression schemes, which is already discussed in Subsection 4.2.5. The solid lines represent the reduced-memory complexity GDIB algorithm applying the one-step compression

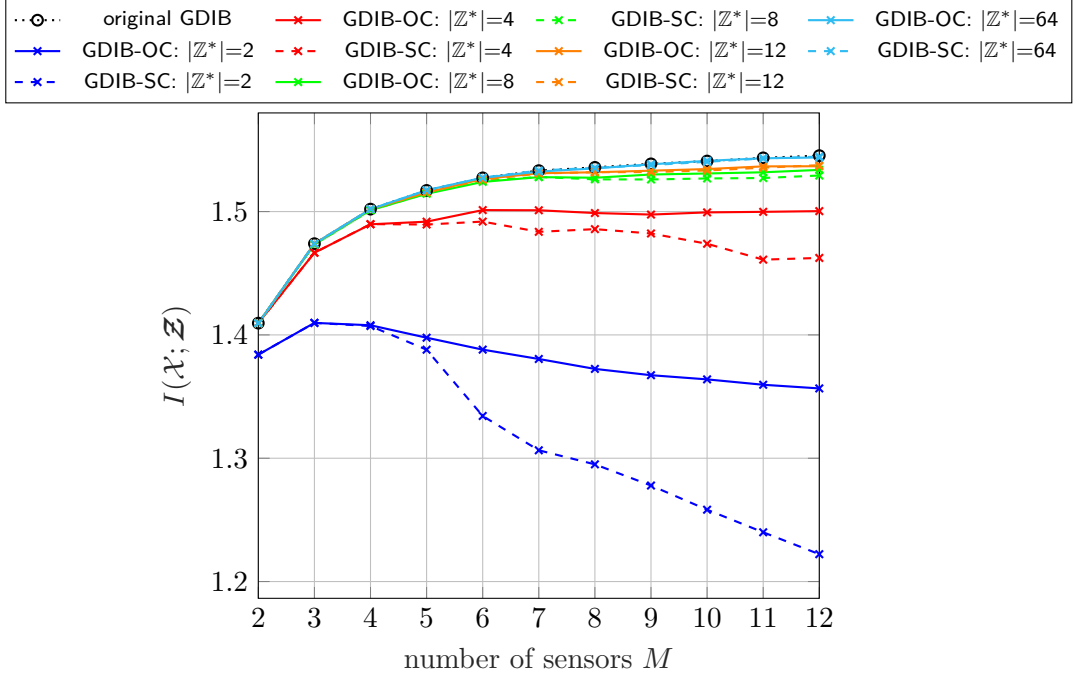


Figure 4.22: Relevant mutual information versus number of sensors; symmetric scenario, sum-rate  $C_{\text{sum}} = 2.5$  bit/s/Hz and  $C_m = \frac{C_{\text{sum}}}{M}$ , SNRs  $\gamma_m = 8$  dB,  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}_m| = 64$  and  $|\mathbb{Z}_m| = 4$ ; One-Step Compression (OC), Sequential Compression (SC)

scheme. The general loss due to compression can be observed by the gap to the uncompressed case. Since the sensors share the same channel and the individual link capacity is equally distributed over all sensors, a larger network leads to a stronger compression. In this case, more information in  $\mathbb{Z}_{<m}^*$  about  $\mathcal{X}$  is required to preserve the relevant mutual information. However, if this is not available, due to compression with low cardinality  $|\mathbb{Z}^*|$ , the overall performance degrades for larger network sizes. As expected, the loss to the uncompressed case decreases using larger cardinalities  $|\mathbb{Z}^*|$ . Of course, a general statement about the required cardinality  $|\mathbb{Z}^*|$  is not meaningful since it depends, i.a., on the network size. However, in the considered setup, a cardinality of  $|\mathbb{Z}^*| = 64$  is sufficient to remove the general compression loss completely and perform as good as in the uncompressed case. The dashed lines represent the reduced-memory complexity GDIB algorithm applying the sequential compression scheme. The introduced accumulation of compression losses of the sequential compression scheme results in a performance degradation compared to the case of a one-step compression. However, by increasing the cardinality  $|\mathbb{Z}^*|$ , this loss can become negligible. In particular, for  $|\mathbb{Z}^*| = 64$ , it disappears completely in the considered setup. In this case, the sequential compression scheme not only performs as good as the one-step compression scheme but also approaches the performance of the uncompressed case. Hence, using an appropriate cardinality  $|\mathbb{Z}^*|$  can reduce the required memory requirement and still approach the performance of the original GDIB algorithm.

In order to highlight the advantage of the reduced-memory complexity GDIB algorithm, the memory requirements to store the single pmf  $p(x|y_{12}, z_{<12}^*)$  used in the calculation of the KL divergence when optimizing the last sensor in a network of  $M = 12$  sensors

$z_{<m}^*$	memory to store $p(x y_{12}, z_{<12}^*)$
uncompressed	8 GiB
$ \mathbb{Z}^*  = 2$	4 KiB
$ \mathbb{Z}^*  = 4$	8 KiB
$ \mathbb{Z}^*  = 8$	16 KiB
$ \mathbb{Z}^*  = 12$	24 KiB
$ \mathbb{Z}^*  = 64$	128 KiB

Figure 4.23: Memory requirements to store  $p(x|y_{12}, z_{<12}^*)$  in a network of  $M = 12$  sensors when optimizing the last one

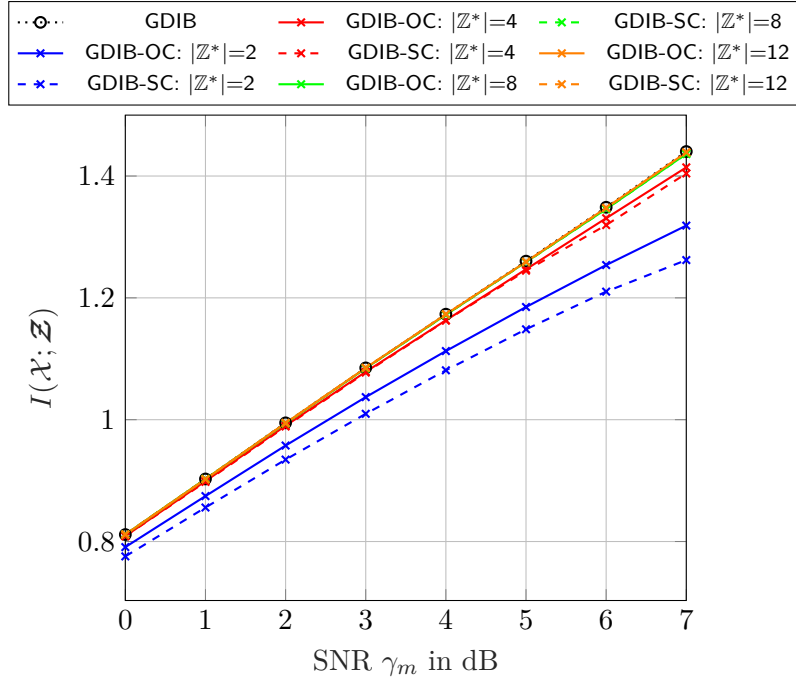


Figure 4.24: Relevant mutual information versus SNR  $\gamma_m$  in dB for symmetric scenario and sum-rate  $C_{\text{sum}} = 2.5$  bit/s/Hz,  $M = 7$ ,  $|\mathbb{Y}_m| = 64$  and  $|\mathbb{Z}_m| = 4$ ; One-Step Compression (OC), Sequential Compression (SC)

are given in Figure 4.23. Using 8 byte for double precision with cardinalities  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}_m| = 64$  and  $|\mathbb{Z}_m| = 4$  it requires  $4 \cdot 64 \cdot 4^{11} \cdot 8$  byte = 8 GiB to store  $p(x|y_M, \mathbf{z}_{<M})$  in the uncompressed case, since for  $|\mathbb{Z}_{<12}| = 4^{11} = 4194304$  holds. Moreover, an increasing cardinality  $|\mathbb{Z}_m|$  would lead to an exponential increase of the memory requirements. Applying the introduced sequential compression scheme reduces these memory requirements significantly. Using a cardinality  $|\mathbb{Z}^*| = 64$  to approach the performance of the uncompressed case, it needs only  $4 \cdot 64 \cdot 64 \cdot 8$  byte = 128 kiB for storing  $p(x|y_M, z_{<M}^*)$ .

Figure 4.24 depicts the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  versus the measurement SNRs  $\gamma_m$  in dB. In this simulation, a network size of  $M = 7$  and the quoted cardinalities have been considered. As before, the sensors share the channel in an orthogonal way and a round-robin fashion, i.e., the sum-rate  $C_{\text{sum}} = 2.5$  bit/s/Hz is equally distributed over all sensors  $C_m = \frac{C_{\text{sum}}}{M}$ . The original GDIB approach is depicted by the black dotted curve. Again, this represents the upper bound. The solid lines represent the reduced-memory complexity GDIB approach using the one-step compression scheme. The dashed lines



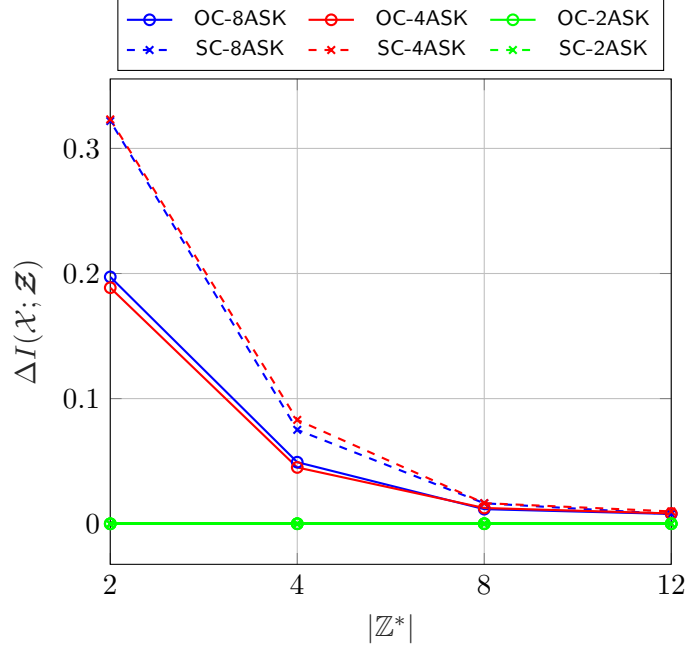


Figure 4.25: Compression loss  $\Delta I(\mathcal{X}; \mathcal{Z})$  versus  $|\mathbb{Z}^*|$ ; Symmetric scenario with  $M = 12$ , sum-rate  $C_{\text{sum}} = 2.5$  bit/s/Hz, SNRs  $\gamma_m = 8$  dB,  $|\mathbb{X}| = \{2, 4, 8\}$ ,  $|\mathbb{Y}_m| = 64$  and  $|\mathbb{Z}_m| = 4$ ; One-Step Compression (OC), Sequential Compression (SC)

represent the sequential compression scheme. Both approaches are given for different cardinalities  $|\mathbb{Z}^*|$ . Naturally, by increasing the SNR, the overall relevant information  $I(\mathcal{X}; \mathcal{Z})$  also increases. Moreover, the loss between the sequential compression scheme and the one-step compression, which shows the accumulated loss due to multiple compression steps, increases for larger SNRs. It can be observed that for larger cardinalities  $|\mathbb{Z}^*|$ , the general difference between compression and no compression becomes smaller. In general, higher measurement SNRs require a larger cardinality  $|\mathbb{Z}^*|$ .

Finally, Figure 4.25 shows the loss in relevant mutual information  $\Delta I(\mathcal{X}; \mathcal{Z}) = I(\mathcal{X}; \mathcal{Z}) - I^c(\mathcal{X}; \mathcal{Z})$  introduced by the different compression schemes versus cardinalities  $|\mathbb{Z}^*|$ . The considered scenario is a network with  $M = 12$  sensors, measurement SNR  $\gamma_m = 8$  dB, and the quoted cardinalities. Again, the sensors share the channel in an orthogonal way and a round-robin fashion, i.e., the sum-rate  $C_{\text{sum}} = 2.5$  bit/s/Hz is equally distributed over all sensors  $C_m = \frac{C_{\text{sum}}}{M}$ . The simulation is done for different alphabets of the relevant signal  $\mathcal{X}$ . As before, the solid lines represent the one-step compression (OC) scheme, while the dashed lines represent the sequential compression (SC) scheme. In the case of a 2-ASK, the compression of  $\mathbf{z}_{<m}$  to  $z_{<m}^*$  does not introduce any error, independent of the compression scheme or the cardinality  $|\mathbb{Z}^*|$ . For a 4-ASK and an 8-ASK, the error becomes more pronounced. As expected by reviewing the previous results, the error for the sequential compression scheme is larger than the one introduced by the one-step compression scheme. However, with increasing cardinality  $|\mathbb{Z}^*|$ , this difference becomes smaller. Moreover, the general loss introduced by compression decreases as well. The considered scenario does not underline the difference between the modulation schemes 4-ASK and 8-ASK. Of course, this might be totally different using a different scenario,

e.g., a higher sum-rate or cardinality  $|\mathbb{Z}_m|$ , which would directly lead to an increase in the achievable relevant information. In this case, the difference between the modulation schemes might be more pronounced.

## 4.4 Distributed Channel Aware Information Bottleneck Algorithm

**A Channel-Aware Optimization Approach:** This section considers a variation of the CEO problem with a logarithmic loss distortion measure where the assumption of perfect transmission over links with specific capacities does not hold anymore. In fact, in real transmission systems with finite-length coding schemes, there will always remain a residual error probability. This setup has already been analyzed in [MA16], where the authors introduced a double max algorithm that aims to maximize the overall relevant mutual information between the relevant signal and the output of all forward channels. In [HWD18d], the authors compared the CAIB approach of [Win14; WMB13] with the double max algorithm in [MA16] and proved their algorithmic equivalence at least for a single sensor and erroneous forward channels. A distributed variant of the CAIB approach has already been proposed in [HWD21] where the authors extend their vector CAIB algorithm proposed in [Has+20] to the multi-terminal case with and without side-information. Moreover, they provided a mathematical analysis of the convergence of the algorithm. This section adapts the previously described GDIB approach defined in Section 4.1 to incorporate a non-perfect forward channel from the sensors to the common receiver. The residual error probability of the forward channel can be modeled by a DMC with a specific transition matrix  $p(\tilde{z}_m|z_m)$ . Therefore, the forward channel has to be taken into account in the optimization process. The general system model can still be illustrated as in Figure 3.17 but with DMCs as forward channels instead of simple capacities  $C_m$ . In fact, it is an extension of the scalar Channel-Aware Information Bottleneck Approach of Section 3.5 to the distributed case. The relevant mutual information  $I(\mathcal{X}; \tilde{\mathcal{Z}}_m)$  for a specific sensor  $m$  is measured between the relevant variable  $\mathcal{X}$  and the output of the forward channel  $\tilde{\mathcal{Z}}_m$ . As the statistic of the DMC is given with cardinalities  $|\mathbb{Z}_m|$  and  $|\tilde{\mathbb{Z}}_m|$ , the compression rate is specified implicitly. Therefore, the focus solely lies on maximizing the relevant mutual information. Inspired by the greedy optimization approach derived in Section 4.1, the optimization problem for each sensor  $m$  becomes

$$p(z_1|y_1)^* = \arg \max_{p(z_1|y_1)} I(\mathcal{X}; \tilde{\mathcal{Z}}_1) \quad (4.29a)$$

$$\vdots$$

$$p(z_M|y_M)^* = \arg \max_{p(z_M|y_M)} I(\mathcal{X}; \tilde{\mathcal{Z}}_M | \tilde{\mathcal{Z}}_{<M}) . \quad (4.29b)$$

**Algorithmic solution:** The derivation of an algorithmic solution follows similar steps as in the scalar case. For a specific sensor  $m$ , the objective becomes

$$p(z_m|y_m)^* = \arg \max_{p(z_m|y_m)} I(\mathcal{X}; \tilde{\mathcal{Z}}_m | \tilde{\mathcal{Z}}_{<m}). \quad (4.30)$$

With

$$\begin{aligned} I(\mathcal{X}; \tilde{\mathcal{Z}}_m, \tilde{\mathcal{Z}}_m | \tilde{\mathcal{Z}}_{<m}) &= I(\mathcal{X}; \tilde{\mathcal{Z}}_m | \tilde{\mathcal{Z}}_{<m}) + \underbrace{I(\mathcal{X}; \tilde{\mathcal{Z}}_m | \tilde{\mathcal{Z}}_{<m}, \tilde{\mathcal{Z}}_m)}_{=0} \\ &= I(\mathcal{X}; \tilde{\mathcal{Z}}_m | \tilde{\mathcal{Z}}_{<m}) + I(\mathcal{X}; \tilde{\mathcal{Z}}_m | \tilde{\mathcal{Z}}_{\leq m}) \end{aligned} \quad (4.31)$$

the relevant mutual information in (4.30) can be rewritten to

$$\begin{aligned} I(\mathcal{X}; \tilde{\mathcal{Z}}_m | \tilde{\mathcal{Z}}_{<m}) &= I(\mathcal{X}; \tilde{\mathcal{Z}}_m | \tilde{\mathcal{Z}}_{<m}) - I(\mathcal{X}; \tilde{\mathcal{Z}}_m | \tilde{\mathcal{Z}}_{\leq m}) \\ &= I(\mathcal{X}; \mathcal{Y}_m | \tilde{\mathcal{Z}}_{<m}) - I(\mathcal{X}; \mathcal{Y}_m | \tilde{\mathcal{Z}}_{<m}, \tilde{\mathcal{Z}}_m) \\ &\quad - I(\mathcal{X}; \mathcal{Y}_m | \tilde{\mathcal{Z}}_{\leq m}) + I(\mathcal{X}; \mathcal{Y}_m | \tilde{\mathcal{Z}}_{\leq m}, \tilde{\mathcal{Z}}_m) \\ &\stackrel{(a)}{=} I(\mathcal{X}; \mathcal{Y}_m | \tilde{\mathcal{Z}}_{<m}) - I(\mathcal{X}; \mathcal{Y}_m | \tilde{\mathcal{Z}}_{\leq m}). \end{aligned} \quad (4.32)$$

Equation (a) holds, since the mutual information  $I(\mathcal{X}; \mathcal{Y}_m | \tilde{\mathcal{Z}}_{<m}, \tilde{\mathcal{Z}}_m)$  and  $I(\mathcal{X}; \mathcal{Y}_m | \tilde{\mathcal{Z}}_{\leq m}, \tilde{\mathcal{Z}}_m)$  are identical due to the Markov property. Since only the last term in (4.32) depends on the mapping  $p(z_m|y_m)$  the optimization problem in (4.30) becomes

$$p(z_m|y_m)^* = \arg \min_{p(z_m|y_m)} I(\mathcal{X}; \mathcal{Y}_m | \tilde{\mathcal{Z}}_{\leq m}). \quad (4.33)$$

The mutual information in (4.33) can now be calculated as

$$\begin{aligned} I(\mathcal{X}; \mathcal{Y}_m | \tilde{\mathcal{Z}}_{\leq m}) &= \mathbb{E}_{\mathcal{X}, \mathcal{Y}_m, \tilde{\mathcal{Z}}_{\leq m}} \left[ \log_2 \frac{p(x|y_m, \tilde{\mathbf{z}}_{\leq m})}{p(x|\tilde{\mathbf{z}}_{\leq m})} \right] \\ &\stackrel{(a)}{=} \sum_{y_m} p(y_m) \sum_{\tilde{\mathbf{z}}_{\leq m}} p(\tilde{\mathbf{z}}_{\leq m}|y_m) \sum_x p(x|y_m, \tilde{\mathbf{z}}_{\leq m}) \log_2 \frac{p(x|y_m, \tilde{\mathbf{z}}_{\leq m})}{p(x|\tilde{\mathbf{z}}_{\leq m})} \\ &\stackrel{(b)}{=} \sum_{y_m} p(y_m) \sum_{\tilde{\mathbf{z}}_{\leq m}} \sum_{\mathbf{y}_{<m}} p(\tilde{\mathbf{z}}_{\leq m}|\mathbf{y}_{<m}) p(\mathbf{y}_{<m}|y_m) D_{\text{KL}} [p(x|y_m, \tilde{\mathbf{z}}_{\leq m}) || p(x|\tilde{\mathbf{z}}_{\leq m})] \\ &\stackrel{(c)}{=} \sum_{y_m} p(y_m) \sum_{\tilde{z}_m} p(\tilde{z}_m|y_m) \sum_{\tilde{\mathbf{z}}_{<m}} \sum_{\mathbf{y}_{<m}} p(\tilde{\mathbf{z}}_{<m}|\mathbf{y}_{<m}) p(\mathbf{y}_{<m}|y_m) D_{\text{KL}} [\cdot || \cdot] \\ &= \sum_{y_m} p(y_m) \sum_{z_m} p(z_m|y_m) \\ &\quad \sum_{\tilde{z}_m} p(\tilde{z}_m|z_m) \sum_{\tilde{\mathbf{z}}_{<m}} \sum_{\mathbf{y}_{<m}} p(\tilde{\mathbf{z}}_{<m}|\mathbf{y}_{<m}) p(\mathbf{y}_{<m}|y_m) D_{\text{KL}} [\cdot || \cdot] \\ &= \sum_{y_m} p(y_m) \sum_{z_m} p(z_m|y_m) C_m(y_m, z_m) \end{aligned} \quad (4.34)$$

where  $C_m(y_m, z_m)$  is defined as

$$C_m(y_m, z_m) := \sum_{\tilde{z}_m} p(\tilde{z}_m | z_m) \sum_{\tilde{\mathbf{z}}_{<m}} \sum_{\mathbf{y}_{<m}} p(\tilde{\mathbf{z}}_{<m} | \mathbf{y}_{<m}) p(\mathbf{y}_{<m} | y_m) D_{\text{KL}}[\cdot || \cdot]. \quad (4.35)$$

In (4.34) equation (a),  $p(x|y_m, \tilde{\mathbf{z}}_{<m})$  reveals from the Markov property, since the dimension of  $\tilde{z}_m$  can be skipped. Expending  $p(\tilde{\mathbf{z}}_{<m}|y_m)$  by  $\mathbf{y}_{<m}$  reveals equation (b). Factorizing  $p(\tilde{\mathbf{z}}_{<m}|\mathbf{y}_{<m}) = \prod_{i=1}^m p(\tilde{z}_i|y_i)$  and decoupling the sum over  $\tilde{z}_m$  leads to equation (c). The term  $C_m(y_m, z_m)$  is non-negative and the mutual information in (4.33) is minimized if  $C_m(y_m, z_m)$  is minimized for each value of  $y_m$ . Hence, for each value  $y_m$  the optimal index  $z_m$  can be found by

$$z_m^*(y_m) = \arg \min_{z_m} C(y_m, z_m). \quad (4.36)$$

Having a closer look at the definition of  $C_m(y_m, z_m)$  in (4.35) it becomes obvious that  $p(\mathbf{y}_{<m}|y_m)$  depends on the number of sensors in the network and might cause memory problems for larger networks. This is even more pronounced since the observation  $y_m$  usually needs to be pre-quantized by an appropriate number of clusters which is generally quite large.

The Algorithm 10 describes the pseudo-code for the Channel Aware Greedy Distributed Information Bottleneck (CA-GDIB) approach for a specific sensor  $m$ . It implements basically a slight modification of the scalar CAIB algorithm described in Section 3.5. Naturally, in order to optimize the complete network, the optimization process has to be done for all sensors. The optimization of a specific sensor  $m$  requires knowledge of the quantizer mappings as well as the DMC transition matrices of previously designed sensors. As a result, the algorithm provides a deterministic mapping  $p(z_m|y_m)$  for sensor  $m$ .

**Performance for Different Measurement SNRs:** Figure 4.26 illustrates the overall performance of the CA-GDIB algorithm compared to the original GDIB algorithm for a network size of  $M = 5$  sensors. The DMCs are modeled as symmetric matrices  $|\mathbb{Z}_m| \times |\tilde{\mathbb{Z}}_m|$  with  $|\mathbb{Z}_m| = |\tilde{\mathbb{Z}}_m|$ . For each symbol, the probability of a correct transmission is given as  $1 - \epsilon$ , whereas all other transitions have the probability  $\frac{\epsilon}{N-1}$ . The relevant signal is chosen to be a 4-ASK signal. As mentioned before, the CA-GDIB algorithm needs to store the pmf  $p(\mathbf{y}_{<m}|y_m)$ , which depends on the number of sensors in the network. Naturally, this may cause memory problems for larger networks. Therefore, the observations are pre-quantized with a low cardinality of  $|\mathbb{Y}_m| = 32$ . The output cardinalities are chosen to  $|\mathbb{Z}_m| = 8$ . The optimization algorithms are initialized with a uniform quantization. Note that the trade-off parameter for the original GDIB algorithm is chosen to  $\beta_m = 0$  in order to be able to compare the CA-GDIB algorithm to the original GDIB algorithm. In this case, both result in deterministic mappings.

As a performance measure, the overall relevant mutual information  $I(\mathcal{X}; \tilde{\mathcal{Z}})$  is depicted for different measurement signal-to-noise ratios  $\gamma_m$ . The gray-shaded area illustrates the non-achievable region since  $I(\mathcal{X}; \mathcal{Y}) \geq I(\mathcal{X}; \tilde{\mathcal{Z}})$  holds due to the data processing inequality. The black dashed-dotted line represents the mutual information  $I(\mathcal{X}; \mathcal{Z})$  achieved by the

---

**Algorithm 10:** Distributed Channel Aware GDIB algorithm
 

---

**input** :  $p(x, y_i), p(\tilde{z}_i|z_i), p(\tilde{z}_i|z_i)\forall i \leq m$   
            $p^{\text{init}}(z_m|y_m), \beta_m, \epsilon$   
**output** :  $p(z_m|y_m) \in \{0, 1\}$

**1 begin**  
     **initialization:**  
          $p(z_m|y_m)^{(0)} \leftarrow p^{\text{init}}(z_m|y_m),$   
          $l \leftarrow 1$

**2**    $p(\tilde{\mathbf{z}}_{<m}|x) = \prod_{i=1}^{m-1} \sum_{y_i} \sum_{z_i} p(\tilde{z}_i|z_i)p(z_i|y_i)p(y_i|x)$   
**3**    $p(\tilde{\mathbf{z}}_{<m}, y_m, x) = p(\tilde{\mathbf{z}}_{<m}|x)p(y_m, x)$   
**4**    $p(x|\tilde{\mathbf{z}}_{<m}, y_m) = p(\tilde{\mathbf{z}}_{<m}, y_m, x) / \sum_x p(\tilde{\mathbf{z}}_{<m}, y_m, x)$   
**5**    $p(\tilde{\mathbf{z}}_{<m}|\mathbf{y}_{<m}) = \prod_{i=1}^{m-1} \sum_{z_i} p(\tilde{z}_i|z_i)p(z_i|y_i)$   
**6**    $p(\mathbf{y}_{\leq m}) = \sum_x p(x) \prod_{i=1}^m p(y_i|x)$   
**7**    $p(\mathbf{y}_{<m}|y_m) = p(\mathbf{y}_{\leq m})/p(y_m)$   
**8**    $p(\tilde{\mathbf{z}}_{<m}|y_m) = \sum_{\mathbf{y}_{<m}} p(\tilde{\mathbf{z}}_{<m}|\mathbf{y}_{<m})p(\mathbf{y}_{<m}|y_m)$

**9 do**  
     **10**    $p(\tilde{z}_m, x)^{(l)} = \sum_{z_m} \sum_{y_m} p(\tilde{z}_m|z_m)p(z_m|y_m)^{(l-1)}p(y_m, x)$   
     **11**    $p(\tilde{\mathbf{z}}_{\leq m}, x)^{(l)} = p(\tilde{\mathbf{z}}_{<m}|x)p(\tilde{z}_m, x)^{(l)}$   
     **12**   // calculate KL divergence  $D_{\text{KL}}(y_m, \tilde{\mathbf{z}}_{\leq m})$   
     **13**    $p(x|\tilde{\mathbf{z}}_{\leq m})^{(l)} = p(\tilde{\mathbf{z}}_{\leq m}, x)^{(l)} / \sum_x p(\tilde{\mathbf{z}}_{\leq m}, x)^{(l)}$   
     **14**    $D_{\text{KL}}(y_m, \tilde{\mathbf{z}}_{\leq m})^{(l)} = \sum_x p(x|\tilde{\mathbf{z}}_{\leq m}, y_m) \cdot \log \frac{p(x|\tilde{\mathbf{z}}_{\leq m}, y_m)}{p(x|\tilde{\mathbf{z}}_{\leq m})^{(l)}}$   
     **15**   // calculate  $C(y_m, z_m)$   
     **16**    $C(y_m, z_m)^{(l)} = \sum_{\tilde{z}_m} p(\tilde{z}_m|z_m) \sum_{\tilde{\mathbf{z}}_{<m}} p(\tilde{\mathbf{z}}_{<m}|y_m) \cdot D_{\text{KL}}(y_m, \tilde{\mathbf{z}}_{\leq m})^{(l)}$   
     **17**   // find minimum of  $C(y_m, z_m)$  for all samples  $y_{m_i} \in \mathbb{Y}_m$   
     **18**   **for**  $y_{m_i} \in \mathbb{Y}_m$  **do**  
     **19**        $z_m^*(y_{m_i})^{(l)} = \arg \min_{z_m} C(y_{m_i}, z_m)^{(l)}$   
     **20**       // update quantizer for specific  $y_{m_i}$   
     **21**        $p(z_m^*(y_{m_i})^{(l)}|y_{m_i})^{(l)} = 1$   
     **22**     $l \leftarrow l + 1$

**23 while**  $D_{\text{JS}}[p^{(l)}(z_m|y_m) || p^{(l-1)}(z_m|y_m)] > \epsilon$ 


---

GDIB algorithm without the influence of the DMC. In a network of  $M = 5$  sensors and an output cardinality of  $|\mathbb{Z}_m| = 8$ , this mutual information is very close to  $I(\mathcal{X}; \mathbf{Y})$ . The solid lines represent the performance of the CA-GDIB algorithm. The dashed lines show the mutual information  $I(\mathcal{X}; \tilde{\mathbf{Z}})$  applying the resulting quantizers of the original GDIB algorithm in a network with a residual error probability. It becomes obvious that the CA-GDIB algorithm outperforms the GDIB algorithm, especially for DMCs with larger error probabilities. Naturally, this is caused by the fact that the CA-GDIB algorithm considers the DMCs within the optimization process. For smaller error probabilities and larger measurement signal-to-noise ratios, there occurs nearly no gap between the performance of the CA-GDIB algorithm and the GDIB algorithm. Moreover, for  $\epsilon = 0.1$  and an SNR of  $\gamma_m > 13$  dB the performance approaches the  $I(\mathcal{X}; \mathbf{Y})$  limit. Hence, the DMC has no influence anymore. Comparing Figure 4.26 to the results of the scalar case in Figure 3.13 it can be observed that gap between  $I(\mathcal{X}; \tilde{\mathbf{Z}})$  and  $I(\mathcal{X}; \mathbf{Z})$  is much smaller in the distributed

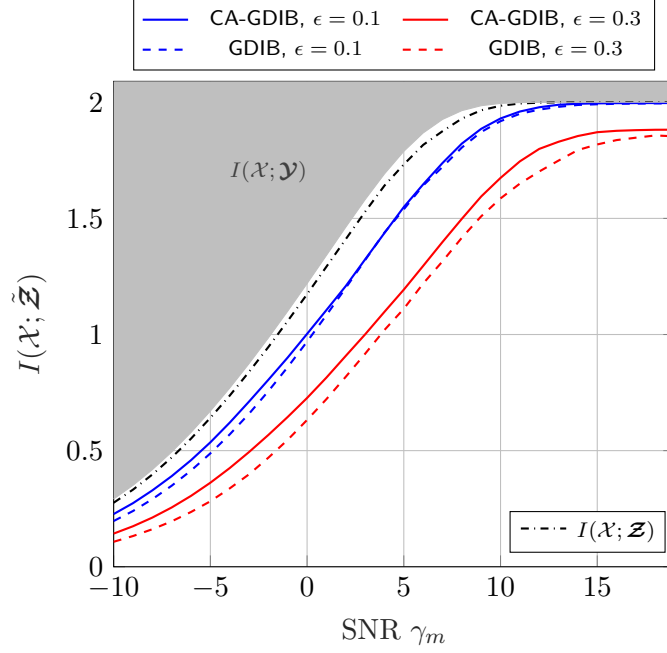


Figure 4.26: Relevant mutual information  $I(\mathcal{X}; \tilde{\mathcal{Z}})$  for different measurement signal-to-noise ratios and an artificially modeled DMC comparing the CA-GDIB algorithm to the GDIB approach for a network size of  $M = 5$ ;  $|\mathcal{X}| = 4$ ,  $|\mathcal{Y}_m| = 32$ ,  $|\mathcal{Z}_m| = 8$

case, especially in higher SNR ranges. This shows that in a network of  $M = 5$  sensors, the negative effect introduced by the DMC is partially compensated by the network size.

**Performance for Different Network Sizes:** Figure 4.27 compares the performance of the CA-GDIB algorithm with the original GDIB algorithm for a varying network size. Therefore, the relevant mutual information  $I(\mathcal{X}; \tilde{\mathcal{Z}})$  is depicted versus the number of sensors in the network  $M$ . As before, the DMCs are modeled as symmetric matrices  $|\mathcal{Z}_m| \times |\tilde{\mathcal{Z}}_m|$  with  $|\mathcal{Z}_m| = |\tilde{\mathcal{Z}}_m|$ . Again, the relevant signal is modeled as a 4-ASK signal. In order to be able to compare the CA-GDIB algorithm to the original GDIB algorithm, the trade-off parameter for the GDIB algorithm has to be chosen to  $\beta_m = 0$  to achieve deterministic mappings. The simulations are done for an SNR of  $\gamma_m = \{3, 8, 12, 15\}$  dB. In all plots, in Figure 4.27, the gray-shaded area illustrates the non-achievable region due to the data processing inequality. Additionally, the mutual information  $I(\mathcal{X}; \mathcal{Z})$  achieved by the original GDIB algorithm without the influence of the DMC is depicted by the black dashed-dotted line. Naturally, it is very close to the optimum  $I(\mathcal{X}; \mathcal{Y})$ , especially for larger network sizes and higher signal-to-noise ratios. The solid line represents the performance of the CA-GDIB algorithm, while the dashed line represents the performance of the original GDIB algorithm. In all plots, it can be observed that the CA-GDIB algorithm outperforms the original GDIB algorithm, especially for DMCs with larger error probabilities. This observation is nearly independent of the network size and is caused by the fact that the CA-GDIB algorithm incorporates the statistics of the DMC into the optimization problem. Only for DMCs with lower error probabilities and larger networks the performance of the original GDIB algorithm approaches the performance of the CA-GDIB algorithm. In this

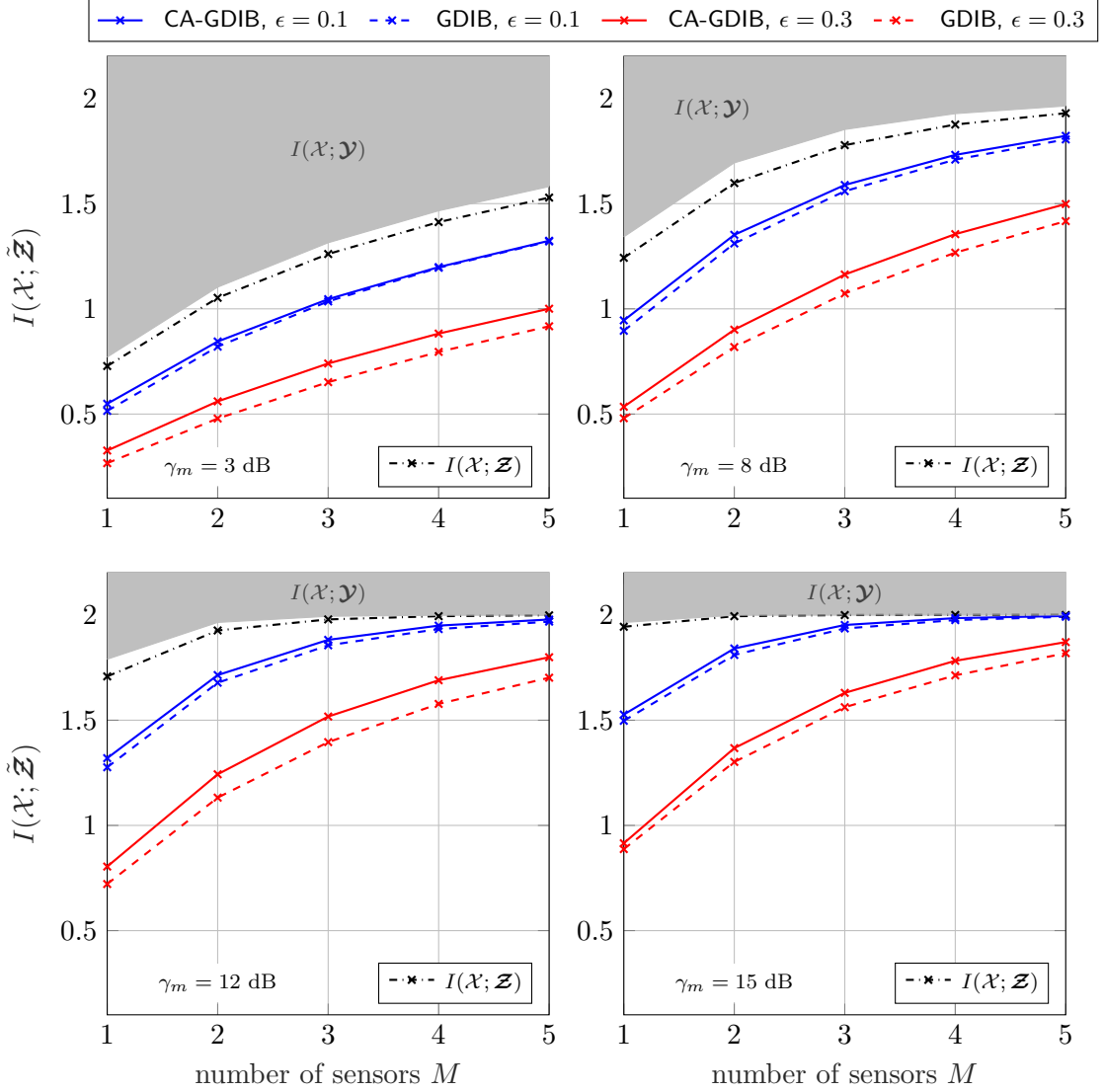


Figure 4.27: Relevant mutual information  $I(\mathcal{X}; \tilde{\mathcal{Z}})$  for network sizes and an artificially modeled DMC comparing the CA-GDIB algorithm to the GDIB approach;  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}_m| = 32$ ,  $|\mathbb{Z}_m| = 8$

case, the DMC has either not a big impact due to larger network sizes or the CA-GDIB algorithm cannot benefit much from incorporating the statistics of the DMC into the optimization problem due to low error probabilities.

**Measurement Channel Mismatch:** Similar to the scalar case in Section 3.5, it has to be analyzed how robust the CA-GDIB algorithm performs for erroneous measurement SNR assumptions. Therefore, a mapping  $\tilde{p}(z_m|y_m)$  is designed for each sensor with the CA-GDIB algorithm using an optimization SNR  $\gamma_m^{\text{opt}}$  with  $\tilde{p}(y_m|x)$ . Afterwards, the mapping  $\tilde{p}(z_m|y_m)$  is applied for a different application SNR  $\gamma_m^{\text{app}}$  with  $p(y_m|x)$ . The resulting relevant mutual information can be determined by

$$\tilde{I}(\mathcal{X}; \tilde{\mathcal{Z}}) = \mathbb{E}_{\mathcal{X}, \tilde{\mathcal{Z}}} \left[ \log_2 \frac{\tilde{p}(\tilde{\mathbf{z}}|x)}{\tilde{p}(\tilde{\mathbf{z}})} \right] \quad (4.37)$$

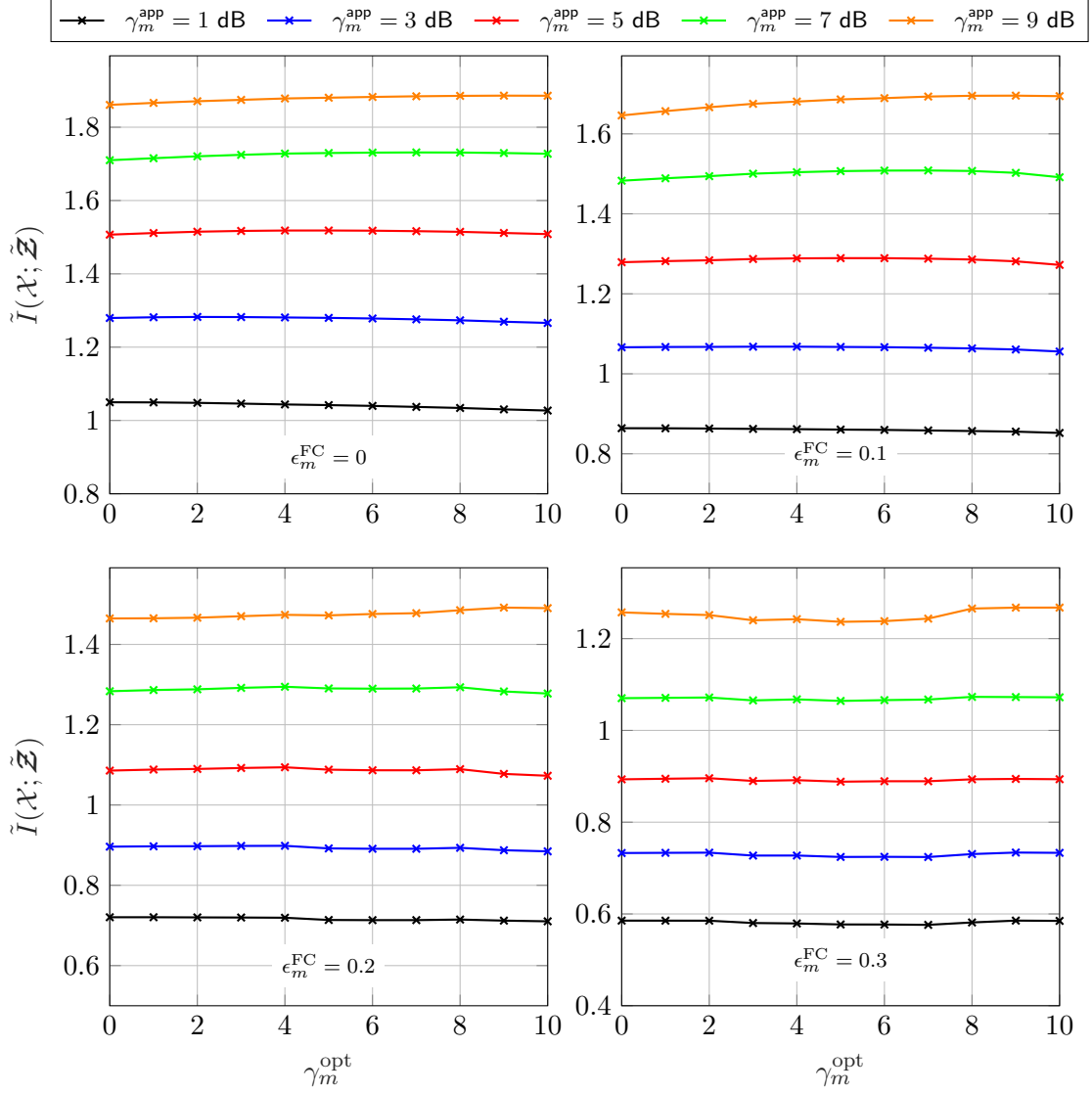


Figure 4.28: Measurement channel mismatch applying the CA-GDIB algorithm with  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}_m| = 512$ ,  $|\mathbb{Z}_m| = 8$  and a network with  $M = 3$  sensors

with

$$\tilde{p}(\tilde{\mathbf{z}}, x) = p(x) \cdot \prod_{m=1}^M \sum_{y_m} \sum_{z_m} p(\tilde{z}_m | z_m) \tilde{p}(z_m | y_m) p(y_m | x) \quad (4.38)$$

and

$$\tilde{p}(\tilde{\mathbf{z}}) = \sum_x \tilde{p}(\tilde{\mathbf{z}}, x). \quad (4.39)$$

Figure 4.28 illustrates the relevant mutual information  $\tilde{I}(\mathcal{X}; \tilde{\mathcal{Z}})$  versus the optimization SNR  $\gamma_m^{\text{opt}}$  for different application SNRs  $\gamma_m^{\text{app}}$  in a symmetric scenario with  $M = 3$  sensors. The different plots show the results for different forward channels  $p(\tilde{z}_m | z_m)$  obtained for a specific  $\epsilon_m^{\text{FC}}$ . The case  $\epsilon_m^{\text{FC}} = 0$  illustrates the performance of the CA-GDIB algorithm for a perfect forward channel introducing no errors. It can be observed that the relevant



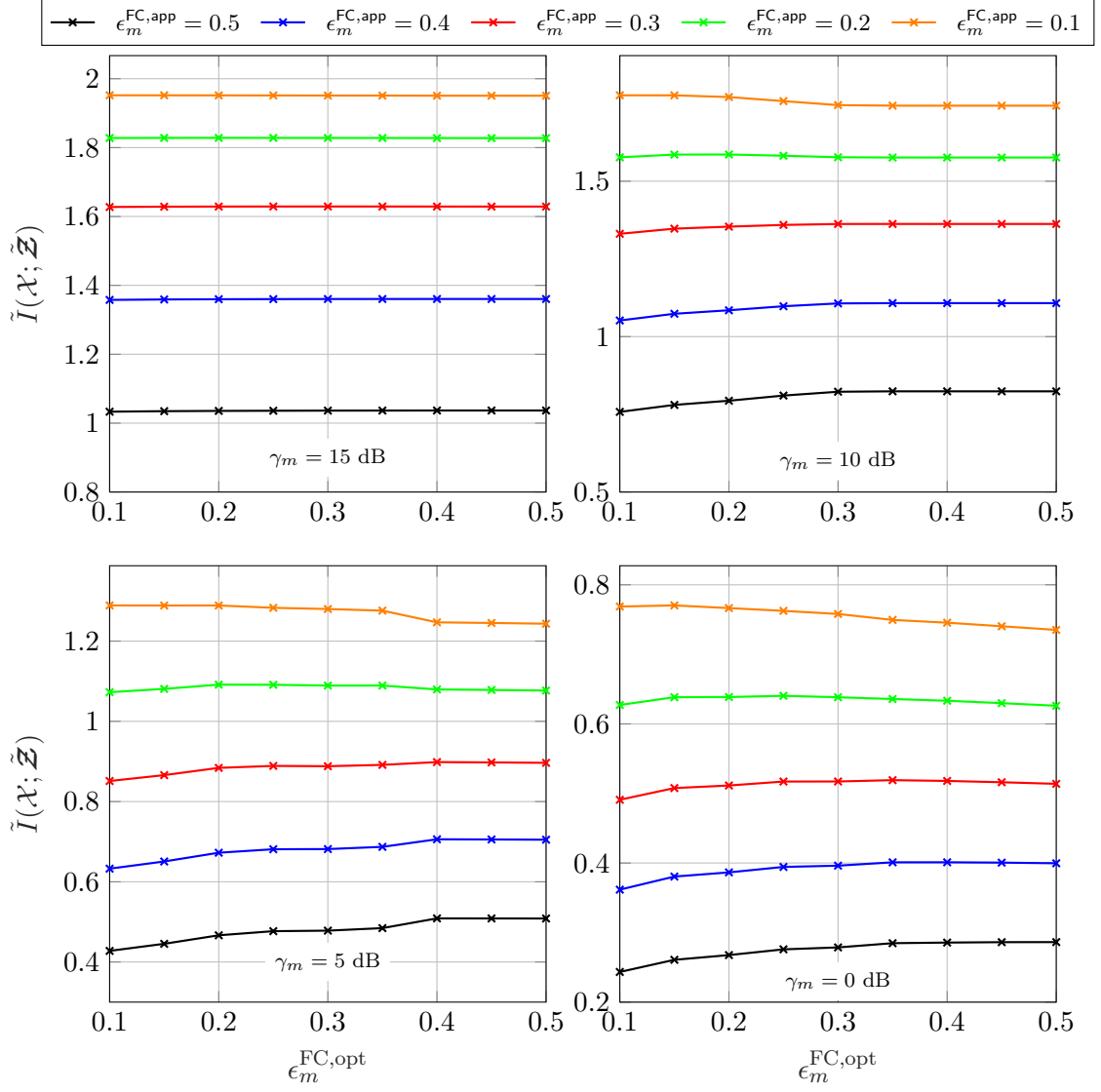


Figure 4.29: Forward channel mismatch applying the CA-GDIB algorithm with  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}_m| = 512$ ,  $|\mathbb{Z}_m| = 8$  and a network with  $M = 3$  sensors

mutual information is nearly constant for all application SNRs and for all forward channels. Therefore, it can be concluded that the CA-GDIB algorithm is quite robust against measurement channel mismatches. Comparing the results in Figure 4.28 to the results of the scalar case in Figure 3.15, it becomes obvious that the distributed algorithm is even more robust in these investigated cases.

**Forward Channel Mismatch:** In order to investigate the robustness of the CA-GDIB against a forward channel mismatch the quantizers  $\tilde{p}(z_m|y_m)$  optimized for forward channels  $\tilde{p}(\tilde{z}_m|z_m)$  defined by  $\epsilon_m^{\text{FC,opt}}$  are applied for different application forward channels  $p(\tilde{z}_m|z_m)$  defined by  $\epsilon_m^{\text{FC,app}}$ .

Figure 4.29 illustrates the relevant mutual information  $\tilde{I}(\mathcal{X}; \tilde{\mathcal{Z}})$  versus the parameter  $\epsilon_m^{\text{FC,opt}}$ . The simulation is done for a symmetric scenario with  $M = 3$  sensors, for different  $\epsilon_m^{\text{FC,app}}$  and different measurement SNRs  $\gamma_m$ . It can be observed that for a large

measurement SNR of  $\gamma_m = 15$  dB, the mismatch has no influence on the relevant mutual information within the investigated range. For each application forward channel  $p(\tilde{z}_m|z_m)$ , the curve is constant. Decreasing the measurement SNR increases the sensitivity on a forward channel mismatch. Although there still occurs a larger plateau where the CA-GDIB algorithm works fine despite a forward channel mismatch, the relevant mutual information  $\tilde{I}(\mathcal{X}; \tilde{\mathcal{Z}})$  decreases when the forward channel differs too much. However, it can be observed that the maximum is always on the position of  $\epsilon_m^{\text{FC,opt}} = \epsilon_m^{\text{FC,app}}$ .

## 4.5 Discussion

This chapter introduced algorithmic approaches for solving the non-cooperative CEO scenario. In particular, a greedy algorithm based on the information bottleneck approach is derived, allowing an offline optimization of each sensor while not exceeding individual forward link capacities [SK21; Ste+21b]. This GDIB algorithm sequentially optimizes each sensor in the network applying the Wyner-Ziv coding principle by exploiting the mappings of previously designed quantizers. In order to fulfill the individual rate constraints of each forward link, a simple bi-section search can be applied. It has been shown that the GDIB optimization can be applied for different relevant signal distributions as long as being discrete or pre-quantized. The GDIB algorithm outperforms independent scalar IB-optimized quantizers in different scenarios. More precisely, it benefits from a growing number of sensors in the network with an increasing or fixed sum-rate. In the case of an increasing sum-rate, the performance of the GDIB algorithm increases much faster than for scalar independent IB optimization. In the case of a fixed sum-rate, the scalar independent IB optimization might even lose performance since the sum-rate is equally distributed on each forward link, while the GDIB performance increases. However, there still occurs a large gap to the fully cooperative CEO scenario. In addition, it turned out that the performance of the GDIB algorithm highly depends on the Wyner-Ziv coding strategy, i.e., different optimization orders result in different spectral efficiencies. As a rule of thumb, it might be beneficial to start the optimization with the sensors belonging to the largest forward link capacity. Similar to the scalar IB approach, the GDIB algorithm is quite robust against measurement SNR mismatches for higher forward link capacities, while it has a higher impact on lower forward link capacities. Here, a mismatch can cause the individual rate constraints to be violated if the application SNR is lower than the optimization SNR. Two approaches are suggested to deal with this behavior. Either a rate back-off could be incorporated in the optimization to ensure the overshoot is still below the actual rate constraint. Or the optimization could be performed for a lower optimization SNR since the overshoot only occurs if the application SNR is lower than the optimization SNR. However, both approaches result in a loss in performance. The GDIB algorithm can be slightly modified to result in deterministic mappings by setting the trade-off parameter to zero. Similar to the scalar IB approach, in this case, the focus solely lies on the preservation of relevant information, while the compression can be adjusted by varying the output cardinality. However, it has been shown that this variation results in quite coarse compression rate steps, which may result in performance losses.

Since the complexity of the GDIB algorithm depends on the number of sensors in the network, the GDIB algorithm suffers from the curse of dimensionality, leading to infeasible optimizations for larger networks. Therefore, a reduced-memory complexity GDIB algorithm has been introduced ensuring feasibility even for larger networks [Ste+21a; Ste+21b] by compressing  $\mathcal{Z}_{<m}$  to  $\mathcal{Z}_{<m}^*$ . It has been shown that the introduced compression schemes may result in a loss in performance compared to the uncompressed case. This can be caused by the IB compression itself or by an accumulation of compression losses due to the sequential compression scheme. However, choosing an appropriate cardinality of  $\mathcal{Z}_{<m}^*$  can remove this loss completely. It has been demonstrated that the sequential compression scheme significantly reduces the memory requirements during the optimization process. Finally, the GDIB algorithm has been modified to incorporate imperfect erroneous forward links. This extension is based on the approach of Winkelbauer in [Win14; WMB13] while still maintaining the greedy optimization structure. This CA-GDIB algorithm has been shown to outperform the original GDIB algorithm for a scenario with a residual error probability on forward links. Moreover, it turned out that the CA-GDIB algorithm is quite robust against measurement and forward channel mismatches.

## Chapter 5

# Distributed Remote Sensing with Partial Cooperation

The original CEO scenario described in Subsection 3.6.1 does not allow the communication among sensors. Although the quantizers of the sensors are jointly designed applying the Wyner-Ziv coding principle in Section 4.1, they are not allowed to exchange any information during run-time. In Chapter 4, the analysis of the non-cooperative GDIB approach revealed that there is a significant gap in relevant mutual information between a fully cooperative CEO scenario and the original non-cooperative CEO scenario. Therefore, this chapter focuses on the question of whether this gap can be closed or at least decreased by allowing the exchange of a certain amount of information among the sensors. The derivations and results were published in [SAK22; SK22]. Section 5.1 introduces the general model, allowing the exchange of information during run-time. Section 5.2 and 5.3 introduce specific protocols defining the way of exchanging this side-information. In particular, Section 5.2 introduces the successive broadcast transmission protocol while Section 5.3 introduces a sequential point-to-point transmission protocol. Finally, Section 5.4 introduces a more theoretical two-phase transmission protocol with artificially created instantaneous side-information.

### 5.1 Partially Cooperative CEO Scenario

In contrast to the original non-cooperative CEO scenario, this system has rarely been investigated in the current literature. Therefore, the complete rate region and upper or lower bounds are still unknown for this scenario. Most works consider jointly Gaussian signals as they allow an analytical treatment. In [PRT04], the authors showed that cooperation among sensors can reduce the compression sum-rate, except for Gaussian relevant variables and an MSE distortion measure. Draper and Wornell considered estimation problems under communication constraints in [DW04]. They propose coding strategies for sensor networks with a specific tree-based structure. Applying the Wyner-Ziv coding principle, solutions for general trees are developed while specific results are given for serial and parallel networks. In [Sim09], the author considered a two-sensor system with a Gaussian source. Simeone investigated two different transmission scenarios. The first consists of orthogonal but rate-limited links between sensors and the common receiver. The second scenario is the Gaussian multiple access channel (MAC). In both cases, cooperation between sensors via rate-limited inter-sensor links leads to substantial gains in terms of the compression sum-rate. In [PSW09], the authors investigated a simple three-node network consisting of an encoder, a helper, and a decoder. It was shown that the provided information by the helper to the encoder is limited by the information given to

the decoder. Any further information does not increase the performance. In [VRP19], the authors introduced a two-way collaborative information bottleneck approach where two sensors observe dependent components and aim to extract information about hidden variables while being able to exchange limited information. For this setup, they characterized the complexity-relevance region. Moreover, they introduced a collaborative distributed information bottleneck approach for a two-sensor scenario in [VVP17; VRP19]. In particular, two sensors observe information and forward it to a third node which tries to learn the relevant information of a hidden process. Again, both sensors can cooperate with each other. For this setup, they introduced inner and outer bounds as well as some characterization of the complexity-relevance region.

The basic idea to improve the performance of the non-cooperative CEO scenario is to allow a limited exchange of information among the sensors via rate-limited inter-sensor links. As this information is exchanged during run-time, it is stated as instantaneous side-information. Moreover, since the exchange is limited due to rate-limited inter-sensor links, the cooperation is done *partially* defining the partially cooperative Chief Executive Officer (pcCEO) scenario. Naturally, the non-cooperative CEO problem represents an extreme case of the pcCEO scenario with zero rate inter-sensor links. The rate-limitation of the links requires the sensors to compress the instantaneous side-information before forwarding it to other sensors. In general, it is assumed that the capacities of the inter-sensor links are larger than the compression rates of the instantaneous side-information  $s_m$ . Within this thesis, only deterministic mappings are considered for the compression of this instantaneous side-information. This can be motivated by the fact that deterministic mappings do not require further lossy compression. Moreover, it allows interpreting the resulting side-information  $s_m$  as indices which help to determine a particular mapping  $y_m \rightarrow s_m$  and  $y_m \rightarrow z_m$  from a list of mappings designed offline in advance. Consequently, the compression rates for the instantaneous side-information can only be adjusted by changing the cardinalities  $|\mathbb{S}_m|$ . Therefore, the inter-sensor links are modeled as bit pipes being able to reliably transmit a specific number of bits.

Following the argumentation of Section 4.1 to derive a formal approach to solve the non-cooperative CEO problem, it becomes obvious that it is hard to define this for the pcCEO problem. To be more precise, the solution for the non-cooperative CEO problem is based on the corresponding inner bound of the rate region given in (3.40). Naturally, this inner bound cannot be applied for partially cooperating sensors since the Markovian structure of the non-cooperative CEO problem in (3.39) does not hold anymore. To the author's knowledge, tight bounds on the rate region are not available for the cooperative case. Therefore, a heuristic approach based on the greedy optimization structure of the GDIB algorithm will be applied to solve the pcCEO scenario, which is not proved to be optimal since the supermodularity of the compression rates of the following optimization problems might not necessarily be fulfilled. Nevertheless, the numerical evaluation of the found solutions demonstrates their usefulness.

In the following, three different inter-sensor communication protocols for exchanging instantaneous side-information are investigated: successive broadcasting, sequential point-to-point exchange, and a two-phase transmission. The two-phase transmission protocol

separates the exchange of instantaneous side-information  $s_m$  and the forwarding of a compressed version of  $z_m$  to the common receiver into two distinct phases. The forwarding to the common receiver starts after the exchange among sensors has been completed, and all sensors have approximately the same amount of side-information. The other two protocols perform the exchange of instantaneous side-information and the forwarding to the common receiver on orthogonal resources. However, they do not separate these operations into distinct phases.

## 5.2 Successive Broadcast Transmission Protocol

The system model using the successive broadcast transmission protocol is given in Figure 5.1. As for the non-cooperative case, each sensor observes a noisy version  $y_m$  of the same relevant signal  $x$  and tries to forward it over capacity-limited links to a common receiver. Therefore, each sensor has to compress the measurements using a specific encoding process. Additionally, each sensor can broadcast instantaneous side-information  $s_m$  to all subsequent sensors. Although a broadcast operation is applied to distribute  $s_m$ , the different sensors do not have the same amount of side-information due to a specific optimization order. This is different to the fcCEO scenario, where the exchange of side-information is performed in a distinct phase. Figure 5.2 illustrates the dependencies among involved random variables using a Markov model. It can be observed that the random variables  $\mathcal{Z}_m$  are coupled via the instantaneous side-information  $\mathcal{S}_{<m}$ . Therefore, given the random variable  $\mathcal{X}$ , different  $\mathcal{Z}_m$  are not independent anymore. This dependence has to be taken into account within the optimization problem and leads to the mathematical structure

$$p(x, \mathbf{y}, \mathbf{z}, \mathbf{s}) = \prod_{m=1}^M p(z_m | y_m, \mathbf{s}_{<m}) p(s_m | y_m, \mathbf{s}_{<m}) p(y_m | x) p(x). \quad (5.1)$$

Sensor  $m$  has access to the instantaneous side-information of all previously designed quantizers  $\mathbf{s}_{<m}$  and can exploit them during its own encoding process. The encoding process of a specific sensor  $m$  is depicted in Figure 5.3. Exploiting the instantaneous side-information of all previously designed quantizers, sensor  $m$  compresses its observation  $y_m$  together with  $\mathbf{s}_{<m}$  to  $z_m$ . This compression can be described by the mapping  $p(z_m | y_m, \mathbf{s}_{<m}) \in [0, 1]$ . As the mapping is generally stochastic, similar to the original CEO problem, a binary encoding and rate-matching step, termed BERM, is required allowing lossless transmission to the common receiver. Additionally, sensor  $m$  produces its instantaneous side-information  $s_m$  using the mapping  $p(s_m | y_m, \mathbf{s}_{<m}) \in \{0, 1\}$ , which shall be deterministic. Both compression steps can be interpreted as a data fusion of the observation and the received instantaneous side-information delivering the corresponding output. Naturally, the compression shall be optimized in order to preserve as much information as possible about the relevant signal  $x$ , motivating the application of the IB principle. This can be illustrated by means of IB graphs in Figure 5.4.

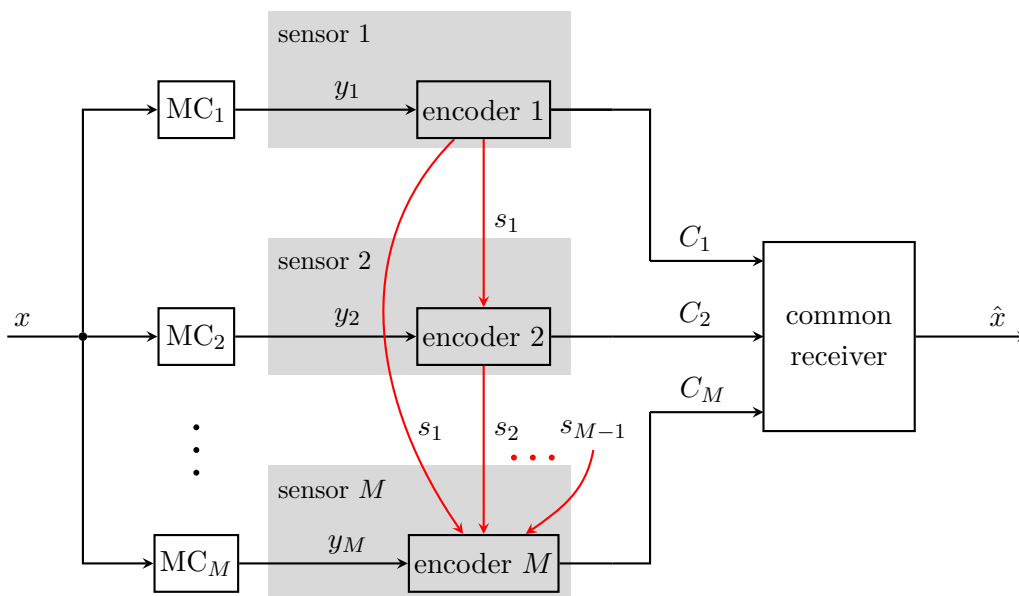


Figure 5.1: System model for the pcCEO scenario using the successive broadcast transmission protocol

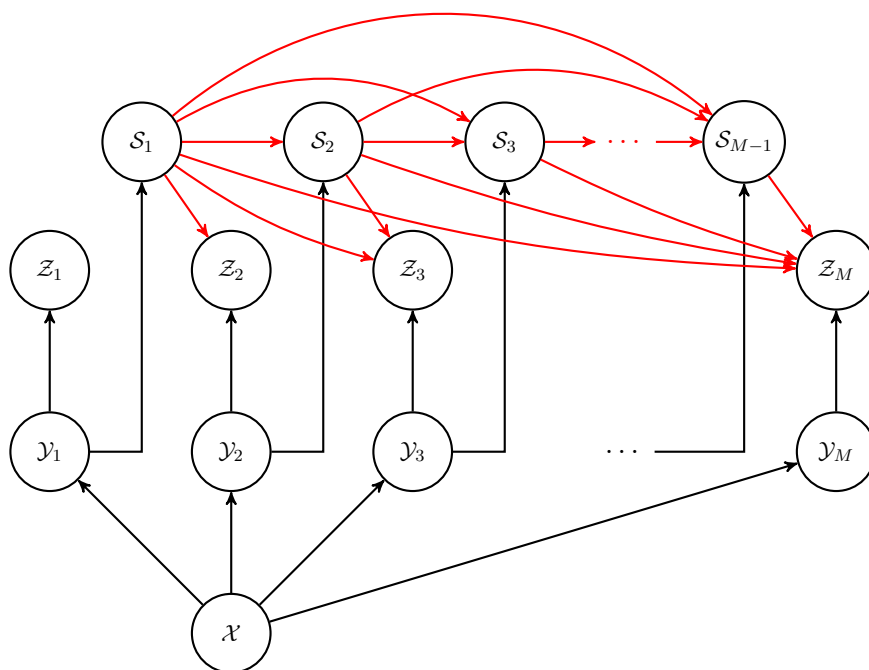


Figure 5.2: Markov model of statistical dependencies for the successive broadcast transmission protocol

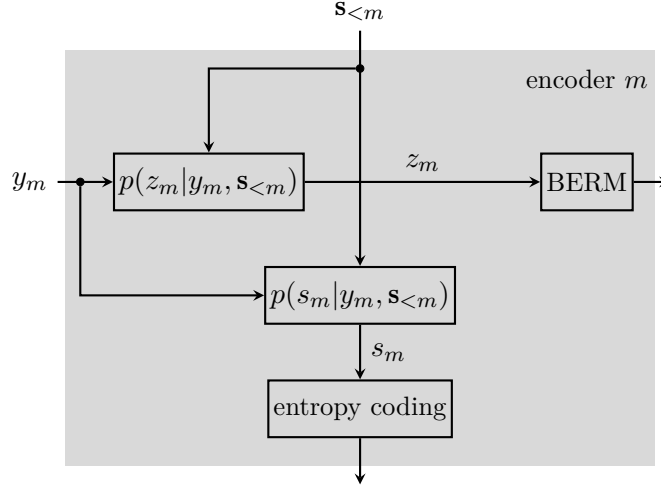
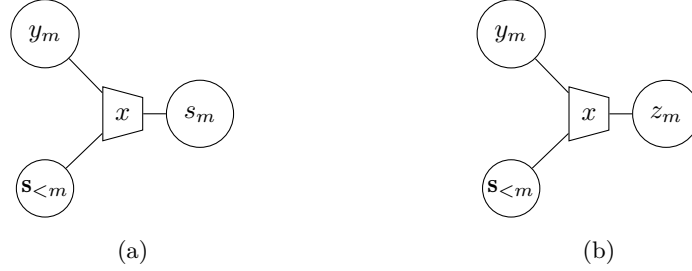


Figure 5.3: Encoding process for successive broadcast transmission protocol


 Figure 5.4: Graphical illustration of IB fusion of two inputs to determine instantaneous side-information  $s_m$  (a) and the quantizer sensor output  $z_m$  (b) for the successive broadcast transmission protocol

### 5.2.1 Generation of Broadcast Side-Information

Inspired by the general GDIB algorithm introduced in Chapter 4, the optimization approach to generate the instantaneous side-information using the broadcast transmission protocol for sensor 1 to  $M - 1$  is performed in a greedy manner.

$$\begin{aligned}
 L_{\text{BC-SIDE}}^{(1)} &= I(\mathcal{X}; \mathcal{S}_1) - \beta I(\mathcal{Y}_1; \mathcal{S}_1) \\
 &\vdots \\
 L_{\text{BC-SIDE}}^{(M-1)} &= I(\mathcal{X}; \mathcal{S}_{M-1} | \mathcal{S}_{<M-1}) - \beta I(\mathcal{Y}_{M-1}; \mathcal{S}_{M-1} | \mathcal{S}_{<M-1})
 \end{aligned}$$

The optimization problem of the first sensor equals individual scalar IB optimization without any side-information. Subsequent sensors combine the instantaneous side-information of all previously transmitting sensors  $\mathbf{s}_{<m}$  with its observation  $y_m$ . Both the relevant mutual information as well as the compression rate of sensor  $m$  are conditioned on  $\mathbf{S}_{<m}$ . This is motivated since broadcasting instantaneous side-information ensures all successive



sensors to have access to all indices  $\mathbf{s}_{<m}$  allowing the application of Wyner-Ziv coding for generating  $s_m$ . For a specific sensor  $m$ , the optimization problem is given as

$$L_{\text{BC-SIDE}}^{(m)}[p(s_m|y_m, \mathbf{s}_{<m})] = I(\mathcal{X}; \mathcal{S}_m | \mathcal{S}_{<m}) - \beta_m I(\mathcal{Y}_m; \mathcal{S}_m | \mathcal{S}_{<m}). \quad (5.2)$$

### Algorithmic Solution

The optimization problem given in (5.2) can be solved by taking the derivative w.r.t. the mapping  $p(s_m|y_m, \mathbf{s}_{<m})$  and equating it to zero. This results in the implicit update equation

$$p(s_m|y_m, \mathbf{s}_{<m}) = \frac{e^{-d_{\beta_m}(y_m, s_m, \mathbf{s}_{<m})}}{\sum_{s_m} e^{-d_{\beta_m}(y_m, s_m, \mathbf{s}_{<m})}} \quad (5.3)$$

with

$$d_{\beta_m}(y_m, s_m, \mathbf{s}_{<m}) := \frac{1}{\beta_m} D_{\text{KL}} [p(x|y_m, \mathbf{s}_{<m}) || p(x|\mathbf{s}_{\leq m})] - \log p(s_m|\mathbf{s}_{<m}). \quad (5.4)$$

Since  $p(x|\mathbf{s}_{\leq m})$  and  $p(s_m|\mathbf{s}_{<m})$  are both depending on the mapping  $p(s_m|y_m, \mathbf{s}_{<m})$  this update equation is implicit and can be solved by an extension of the Blahut-Arimoto algorithm. A detailed derivation of the above solution and the involved pmfs is given in Appendix C.

Since the mapping  $p(s_m|y_m, \mathbf{s}_{<m}) \in \{0, 1\}$  is assumed to be deterministic,  $\beta_m = 0$  is chosen. Following the argumentation in Subsection 3.4.2, the logarithm term in (5.4) can be neglected since the compression rate has no influence anymore. Moreover, the sum in the denominator of the update equation (5.3) is dominated by the smallest KL divergence. Therefore, for each sample  $y_m$  and  $\mathbf{s}_{<m}$ , the mapping  $p(s_m|y_m, \mathbf{s}_{<m})$  tends to zero for all  $s_m$  except for the one with the smallest KL divergence, which tends to one. This leads to

$$s_m^*(y_m, \mathbf{s}_{<m}) = \arg \min_{s_m} D_{\text{KL}} [p(x|y_m, \mathbf{s}_{<m}) || p(x|\mathbf{s}_{\leq m})] \quad (5.5)$$

and the update equation

$$p(s_m|y_m, \mathbf{s}_{<m}) = \begin{cases} 1 & \text{for } s_m(y_m, \mathbf{s}_{<m}) = s_m^*(y_m, \mathbf{s}_{<m}) \\ 0 & \text{else.} \end{cases} \quad (5.6)$$

The extended Blahut-Arimoto algorithm to obtain the instantaneous side-information in the broadcast case for sensor  $m$  and  $\beta_m = 0$  is given in Algorithm 11. The pmf  $p(\mathbf{s}_{<m}|x)$  is given as an input and has already been calculated recursively during the optimization of previous sensors. Lines 2 and 3 calculate some required pmfs which do not depend on the sensor currently being optimized. Lines 5 to 8 calculate the KL divergence of (5.4) as well as the required pmfs. In lines 9 to 14, the minimum of this KL divergence is determined, which defines the position of the ones in the mapping  $p(s_m|y_m, \mathbf{s}_{<m})$ . As in the original GDIB algorithm, the optimization is done until subsequent quantizer mappings do not

---

**Algorithm 11:** Extended Blahut-Arimoto algorithm to obtain instantaneous side-information for the successive broadcasting protocol

---

```

input      :  $m, p^{\text{init}}(s_m|y_m, \mathbf{s}_{<m}), p(y_m, x), \epsilon$ 
               recursively calculated inputs from previous sensor optimizations:
                $p(\mathbf{s}_{<m}|x)$ 
output    :  $p(s_m|y_m, \mathbf{s}_{<m}) \in \{0, 1\}, p(\mathbf{s}_{\leq m}|x)$ 
1 begin
   initialization:
        $p(s_m|y_m, \mathbf{s}_{<m})^{(0)} \leftarrow p^{\text{init}}(s_m|y_m, \mathbf{s}_{<m}),$ 
        $l \leftarrow 1$ 
2    $p(y_m, \mathbf{s}_{<m}, x) = p(y_m|x)p(\mathbf{s}_{<m}|x)p(x)$ 
3    $p(x|y_m, \mathbf{s}_{<m}) = p(y_m, \mathbf{s}_{<m}, x) / \sum_x p(y_m, \mathbf{s}_{<m}, x)$ 
4   do
5       // KL-Divergence  $D_{\text{KL}}(y_m, \mathbf{s}_{\leq m})$  of (5.4)
6        $p(\mathbf{s}_{\leq m}|x)^{(l)} = \sum_{y_m} p(s_m|y_m, \mathbf{s}_{<m})^{(l-1)} p(y_m|x) p(\mathbf{s}_{<m}|x)$ 
7        $p(x|\mathbf{s}_{\leq m})^{(l)} = p(\mathbf{s}_{\leq m}|x)^{(l)} p(x) / \sum_x p(\mathbf{s}_{\leq m}|x)^{(l)} p(x)$ 
8        $D_{\text{KL}}(y_m, \mathbf{s}_{\leq m})^{(l)} = \sum_x p(x|y_m, \mathbf{s}_{\leq m}) \cdot \log \frac{p(x|y_m, \mathbf{s}_{\leq m})}{p(x|\mathbf{s}_{\leq m})}$ 
9       // find minimum of  $D_{\text{KL}}(y_m, \mathbf{s}_{\leq m})$  for all samples  $y_m \in \mathbb{Y}_m$  and
        $\mathbf{s}_{<m} \in \mathbb{S}_{<m}$ 
10      for  $y_i \in \mathbb{Y}_m$  do
11          for  $\mathbf{s}_j \in \mathbb{S}_{<m}$  do
12               $s_m^*(y_i, \mathbf{s}_j)^{(l)} = \arg \min_{s_m} D_{\text{KL}}(y_m, \mathbf{s}_{\leq m})^{(l)}$ 
13              // update quantizer for specific  $y_i$  and  $\mathbf{s}_j$ 
14               $p(s_m^*(y_i, \mathbf{s}_j)^{(l)}|y_i, \mathbf{s}_j)^{(l)} = 1$ 
15           $l \leftarrow l + 1$ 
16      while  $D_{\text{JS}}[p^{(l)}(s_m|y_m, \mathbf{s}_{<m}) || p^{(l-1)}(s_m|y_m, \mathbf{s}_{<m})] > \epsilon$ 
17      // update  $p(\mathbf{s}_{\leq m}|x)$  serving as an input for the successive sensor
18       $p(\mathbf{s}_{\leq m}|x) = \sum_{y_m} p(s_m|y_m, \mathbf{s}_{<m}) p(y_m|x) p(\mathbf{s}_{<m}|x)$ 

```

---

change significantly anymore. After that, the mapping  $p(\mathbf{s}_{\leq m}|x)$  has to be updated to be used as an input for subsequent sensors.

**Evolution of Instantaneous Side-Information Available at Sensors:** Figure 5.5 illustrates the amount of instantaneous side-information available at the different sensors in the network. Therefore, a network of size  $M = 6$  is considered using the broadcast transmission protocol. The relevant mutual side-information  $I(\mathcal{X}; \mathbf{S}_{\leq m})$  is depicted versus the sensor index  $m$  for different cardinalities  $|\mathbb{S}_m|$  and different SNRs  $\gamma_m$ . The relevant signal is chosen to be a 4-ASK signal. It can be observed that the resolution and the quality of instantaneous side-information available at sensor  $m$  increases with growing  $m$ . Naturally, the amount of information a sensor can contribute to  $I(\mathcal{X}; \mathbf{S}_{\leq m})$  decreases with each additional sensor. This general behavior is independent of the cardinality  $|\mathbb{S}_m|$  or the SNR. The largest gain can be observed between the cardinalities  $|\mathbb{S}_m| = 2$  and  $|\mathbb{S}_m| = 4$  since one bit is not enough to represent a 4-ASK signal with  $|\mathbb{X}| = 4$ . Therefore, increasing

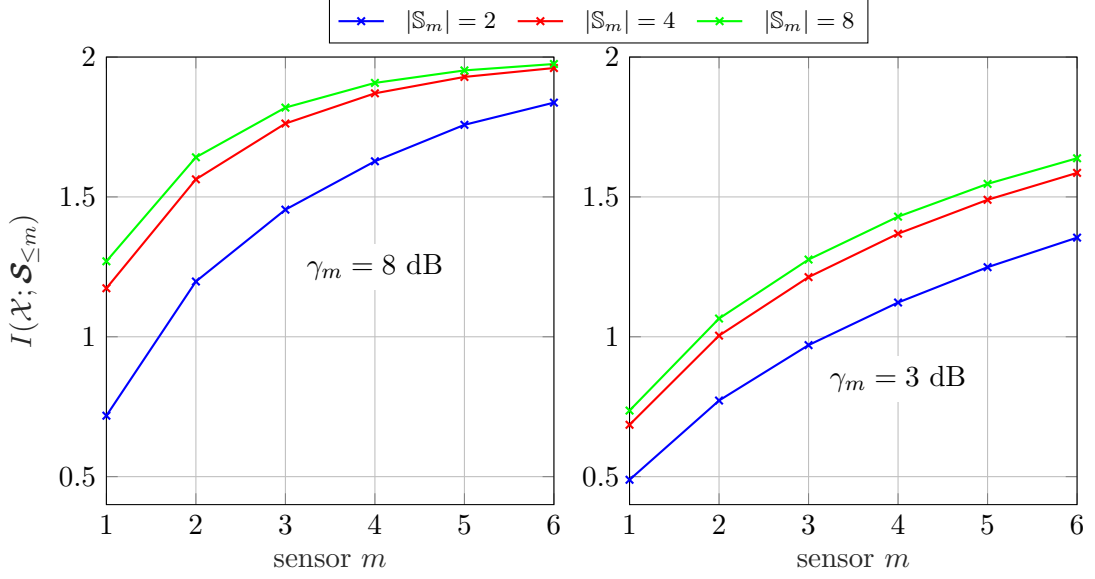


Figure 5.5: Evolution of side-information  $I(\mathcal{X}; \mathbf{S}_{\leq m})$  for sensor  $m$  in a network with  $M = 6$  sensors and different cardinalities  $|\mathbb{S}_m|$  for the successive broadcasting protocol;  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}_m| = 64$

the cardinality to  $|\mathbb{S}_m| = 8$  only results in a smaller gain. Note that this highly depends on the relevant signal  $\mathcal{X}$  and can not be generalized. It has to be mentioned that even if the instantaneous side-information only contains a single bit, i.e.,  $|\mathbb{S}_m| = 2$ , the relevant mutual information  $I(\mathcal{X}; \mathbf{S}_{\leq m})$  can become larger than 1 bit/s/Hz for later sensors. This is caused by  $\mathbf{S}_{\leq m}$  including the instantaneous side-information of all previous sensors such that  $|\mathbb{S}_{\leq m}| = |\mathbb{S}_m|^m$  if  $|\mathbb{S}_m|$  is equal for all sensors.

### 5.2.2 Generate Information Forwarded to the Common Receiver

The instantaneous side-information designed in the previous subsection can now be exploited in an optimization approach to maximize the relevant mutual information  $I(\mathcal{X}; \mathbf{Z})$  while still fulfilling the individual rate constraints of each sensor. Therefore, the original GDIB algorithm is modified, defining the Greedy Distributed Information Bottleneck-Broadcast Side-Information (GDIB-BC) algorithm. Considering the encoding process depicted in Figure 5.3, the quantizer which produces the output  $z_m$  fuses  $y_m$  and  $\mathbf{s}_{< m}$  to a single output  $z_m$ . The optimization problem can be defined as

$$\begin{aligned} L_{\text{GDIB-BC}}^{(1)} &= I(\mathcal{X}; \mathbf{Z}_1) - \beta_1 I(\mathcal{Y}_1; \mathbf{Z}_1) \\ &\vdots \\ L_{\text{GDIB-BC}}^{(M)} &= I(\mathcal{X}; \mathbf{Z}_M | \mathbf{Z}_{< M}) - \beta_M I(\mathcal{Y}_M, \mathbf{S}_{< M}; \mathbf{Z}_M | \mathbf{Z}_{< M}). \end{aligned}$$

Naturally, the first sensor has no side-information at all, and the optimization problem reduces to the scalar IB problem. Subsequent sensors exploit both the quantizer mappings of all previously designed sensors by Wyner-Ziv coding as well the instantaneous side-information received via broadcasting of all previous sensors. These optimization problems

are very similar to the original GDIB approach. The main difference lies in the definition of the compression rate  $I(\mathcal{Y}_m, \mathbf{S}_{<m}; \mathcal{Z}_m | \mathcal{Z}_{<m})$  which emerges from the combination of the observation  $y_m$  and the instantaneous side-information  $\mathbf{s}_{<m}$ . The optimization problem of sensor  $m$  becomes

$$L_{\text{GDIB-BC}}^{(m)}[p(z_m | y_m, \mathbf{s}_{<m})] = I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{<m}) - \beta_m I(\mathcal{Y}_m, \mathbf{S}_{<m}; \mathcal{Z}_m | \mathcal{Z}_{<m}). \quad (5.7)$$

### Algorithmic Solution

The optimization problem given in (5.7) can be solved by taking the derivative w.r.t. the mapping  $p(z_m | y_m, \mathbf{s}_{<m})$  and equating it to zero, leading to an implicit update equation

$$p(z_m | y_m, \mathbf{s}_{<m}) = \frac{e^{-d_{\beta_m}(y_m, z_m, \mathbf{s}_{<m})}}{\sum_{z_m} e^{-d_{\beta_m}(y_m, z_m, \mathbf{s}_{<m})}} \quad (5.8)$$

with

$$d_{\beta_m}(y_m, z_m, \mathbf{s}_{<m}) := \mathbb{E}_{\mathcal{Z}_{<m} | y_m, \mathbf{s}_{<m}} \left[ \frac{1}{\beta_m} D_{\text{KL}} [p(x | y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) || p(x | \mathbf{z}_{\leq m})] - \log p(z_m | \mathbf{z}_{<m}) \right]. \quad (5.9)$$

Since  $p(z_m | \mathbf{z}_{<m})$  and  $p(x | \mathbf{z}_{\leq m})$  are dependent on the desired mapping  $p(z_m | y_m, \mathbf{s}_{<m})$ , (5.8) is an implicit equation. Using a Blahut-Arimoto-like algorithm, this implicit update equation can be solved, resulting in local optima. A detailed derivation of the above solution as well as a way to calculate the required pmfs taking into account the statistical dependencies from (5.1) is given in Appendix D.

The modified Blahut-Arimoto algorithm to design the quantizer of sensor  $m$  for a specific  $\beta_m$  is given in Algorithm 12. The input pmf  $p(y_{m-1}, \mathbf{s}_{<m-1}, \mathbf{z}_{<m-1}, x)$  has been calculated during the optimization of previous sensors. Of course, this simplifies for the first sensor since it has no predecessor. Lines 2 to 4 determine required pmfs, which do not depend on the sensor currently being optimized. The KL divergence in (5.9) is determined in lines 6 to 9. The whole statistical distance of (5.9) is calculated in lines 10 to 13. It is used to update the quantizer mapping  $p(z_m | y_m, \mathbf{s}_{<m})$  of sensor  $m$ . This procedure is repeated until no significant changes in the quantizer mapping occur anymore. The algorithm returns the optimized mapping  $p(z_m | y_m, \mathbf{s}_{<m})$  as well as the pmf  $p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}, x)$  which is used as an input for the successive sensor.

The flowchart, which describes the original GDIB algorithm in Figure 4.1, is also valid for partially cooperating sensors using the broadcast transmission protocol. The difference is mainly the extended Blahut-Arimoto algorithm. Hence, Algorithm 12 has to be performed for each sensor. In order to find the rate fulfilling parameter  $\beta_m$ , which determines the compression rate at sensor  $m$ , a bisection search can be applied. Note that in the case of broadcasting sensors, the compression rate is calculated as  $I(\mathcal{Y}_m, \mathbf{S}_{<m}; \mathcal{Z}_m | \mathcal{Z}_{<m})$ . Similar to the original GDIB algorithm, the optimization has to be performed for all possible permutations of the optimization order. Note that in contrast to the original GDIB algorithm, the GDIB-BC does not necessarily find extreme points of the solution space corresponding

---

**Algorithm 12:** Extended Blahut-Arimoto algorithm for the successive broadcasting protocol

---

**input** :  $m, p^{\text{init}}(z_m|y_m, \mathbf{s}_{<m}), p(y_i, x), p(s_i|y_i, \mathbf{s}_{<i}) \forall i \leq m, \beta_m, \epsilon$

recursively calculated input from previous sensor optimizations:  
 $p(y_{m-1}, \mathbf{s}_{<m-1}, \mathbf{z}_{<m-1}, x)$

**output** :  $p(z_m|y_m, \mathbf{s}_{<m}) \in [0, 1], p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}, x)$

**1 begin**

**initialization:**

$p(z_m|y_m, \mathbf{s}_{<m})^{(0)} \leftarrow p^{\text{init}}(z_m|y_m, \mathbf{s}_{<m}),$   
 $l \leftarrow 1$

**2**  $p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}, x) = \sum_{y_{m-1}} p(z_{m-1}|y_{m-1}, \mathbf{s}_{<m-1})p(s_{m-1}|y_{m-1}, \mathbf{s}_{<m-1}) \cdot$   
 $p(y_m|x)p(y_{m-1}, \mathbf{s}_{<m-1}, \mathbf{z}_{<m-1}, x)$

**3**  $p(x|y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) = p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}, x) / \sum_x p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}, x)$

**4**  $p(\mathbf{z}_{<m}|y_m, \mathbf{s}_{<m}) = \sum_x p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}, x) / \sum_x \sum_{\mathbf{z}_{<m}} p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}, x)$

**5 do**

**6** // KL-Divergence  $D_{\text{KL}}(y_m, \mathbf{s}_{<m}, \mathbf{z}_{\leq m})$  of (5.9)

$p(\mathbf{z}_{\leq m}, x)^{(l)} = \sum_{\mathbf{s}_{<m}} \sum_{y_m} p(z_m|y_m, \mathbf{s}_{<m})^{(l-1)}p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}, x)$

$p(x|\mathbf{z}_{\leq m})^{(l)} = p(\mathbf{z}_{\leq m}, x)^{(l)} / \sum_x p(\mathbf{z}_{\leq m}, x)^{(l)}$

$D_{\text{KL}}(y_m, \mathbf{s}_{<m}, \mathbf{z}_{\leq m})^{(l)} = \sum_x p(x|y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) \cdot \log \frac{p(x|y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m})}{p(x|\mathbf{z}_{\leq m})^{(l)}}$

**10** // distance  $d_{\beta_m}(y_m, z_m, \mathbf{s}_{<m})$  (5.9)

$p(\mathbf{z}_{\leq m})^{(l)} = \sum_x p(\mathbf{z}_{\leq m}, x)^{(l)}$

$p(z_m|\mathbf{z}_{<m})^{(l)} = p(\mathbf{z}_{\leq m})^{(l)} / \sum_{z_m} p(\mathbf{z}_{\leq m})^{(l)}$

$d_{\beta_m}(z_m, y_m, \mathbf{s}_{<m})^{(l)} =$   
 $\sum_{\mathbf{z}_{<m}} p(\mathbf{z}_{<m}|y_m, \mathbf{s}_{<m}) \cdot \left[ \frac{1}{\beta_m} D_{\text{KL}}(y_m, \mathbf{s}_{<m}, \mathbf{z}_{\leq m})^{(l)} - \log p(z_m|\mathbf{z}_{<m})^{(l)} \right]$

**14** // update quantizer  $p(z_m|y_m, \mathbf{s}_{<m})$

$p(z_m|y_m, \mathbf{s}_{<m})^{(l)} = \frac{1}{\sum_z e^{-d_{\beta_m}(y_m, z_m, \mathbf{s}_{<m})^{(l)}}} e^{-d_{\beta_m}(y_m, z_m, \mathbf{s}_{<m})^{(l)}}$

$l \leftarrow l + 1$

**17 while**  $D_{\text{JS}}[p^{(l)}(z_m|y_m, \mathbf{s}_{<m}) || p^{(l-1)}(z_m|y_m, \mathbf{s}_{<m})] > \epsilon$

---

to different optimization orders. Therefore, solutions found for specific optimization orders might not be optimal. Considering the involved pmfs in the above optimization approach, it becomes obvious that the optimization might suffer from the curse of dimensionality for larger networks since they grow exponentially with the network size.

**Influence of Network Size:** Figures 5.6 and 5.7 illustrate the overall performance of the pcCEO scenario using the successive broadcasting protocol for different network sizes. Therefore, the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  is depicted versus the number of sensors  $M$  in the network. According to the data-processing inequality, the absolute maximum can be defined by  $I(\mathcal{X}; \mathcal{Z}) \leq I(\mathcal{X}; \mathcal{Y})$ . Hence, the gray-shaded area represents the non-achievable region. For this simulation, the same scenario as in Figure 4.5 and 4.6 is considered, i.e., a scenario where all sensors share the same channel to the common re-

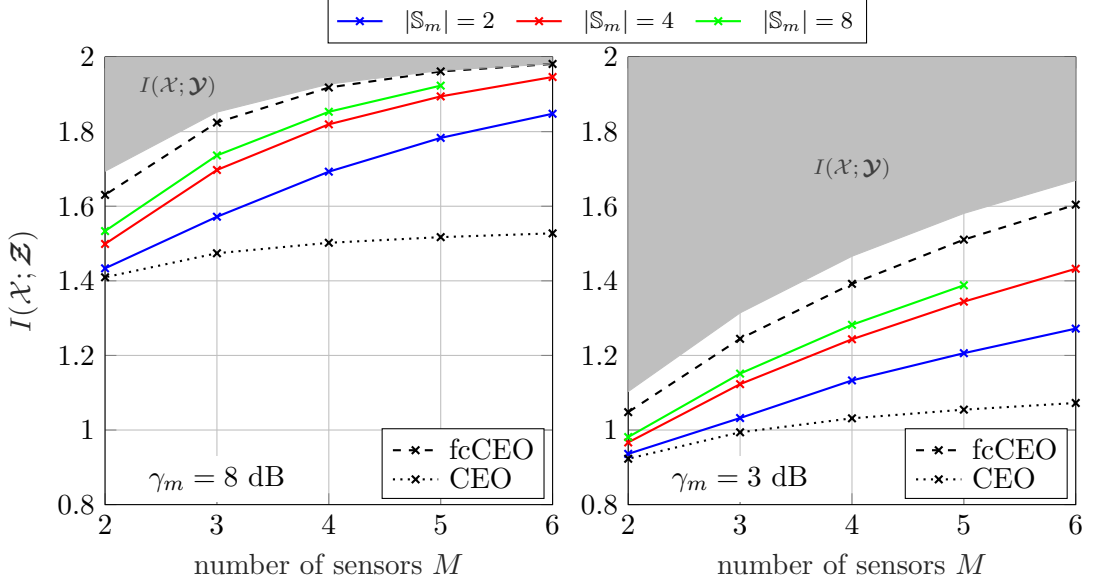


Figure 5.6: Relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  versus the network size for a fixed sum-rate of  $C_{\text{sum}} = 2.5$  bit/s/Hz and  $C_m = \frac{C_{\text{sum}}}{M}$  using the successive broadcasting protocol with different cardinalities  $|\mathcal{S}_m|$ ;  $|\mathcal{X}| = 4$ ,  $|\mathcal{Y}_m| = 64$ ,  $|\mathcal{Z}_m| = 4$

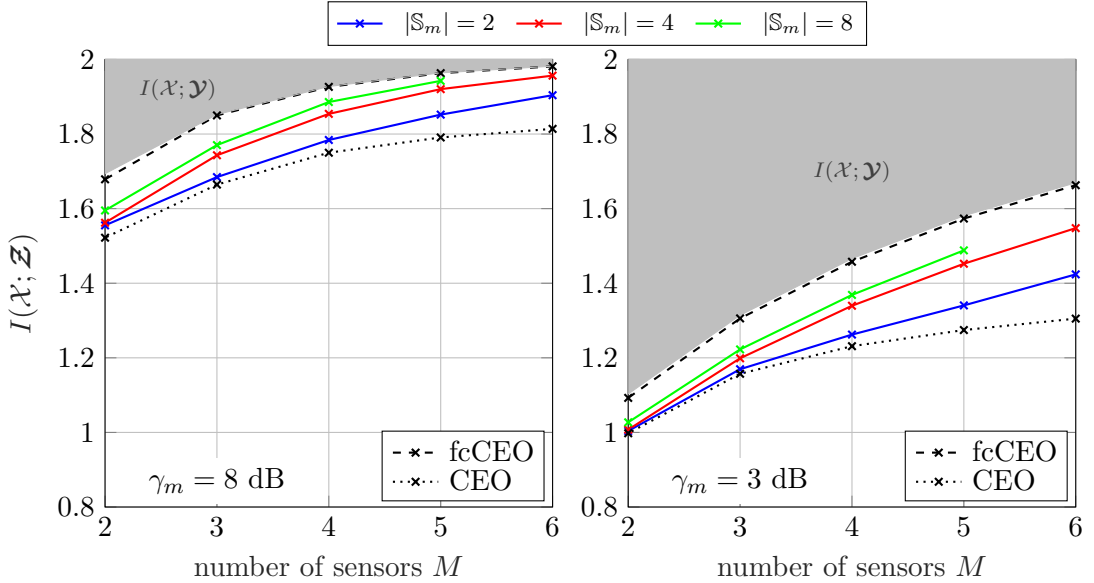


Figure 5.7: Relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  versus the network size for a fixed sum-rate of  $C_{\text{sum}} = 4$  bit/s/Hz and  $C_m = \frac{C_{\text{sum}}}{M}$  using the successive broadcasting protocol with different cardinalities  $|\mathcal{S}_m|$ ;  $|\mathcal{X}| = 4$ ,  $|\mathcal{Y}_m| = 64$ ,  $|\mathcal{Z}_m| = 4$

ceiver in an orthogonal way and a round-robin fashion. The sum-rate is equally distributed among all forward links from the sensors to the common receiver  $C_{\text{sum}} = \sum_{m=1}^M C_m$  with  $C_m = \frac{C_{\text{sum}}}{M}$ . The performance of the pcCEO scenario is compared to the non-cooperative CEO scenario where no communication among sensors is possible and the fcCEO scenario of Subsection 3.6.2. These curves represent the lower and upper bounds, respectively. Note that a detailed performance analysis of the non-cooperative CEO scenario is given in Section 4.2. It can be observed that by increasing the number of sensors in the network,

the performance, i.e., the overall relevant information  $I(\mathcal{X}; \mathcal{Z})$ , also increases. In general, allowing the sensors to communicate via inter-sensor links using the successive broadcasting protocol increases the overall performance compared to the non-cooperative CEO scenario. This holds independent of the cardinality of the instantaneous side-information  $|\mathbb{S}_m|$  and can be explained by sensors being able to exploit the information  $\mathbf{s}_{<m}$  has about the relevant variable  $x$ . Naturally, the amount of information  $\mathbf{s}_{<m}$  has about  $x$  increases for larger network sizes. Therefore, the difference to the non-cooperative CEO scenario increases. Moreover, increasing the cardinality  $|\mathbb{S}_m|$  also improves  $I(\mathcal{X}; \mathcal{S}_{<m})$ , as observed in Figure 5.5, which is why the overall performance benefits from larger cardinalities. However, it can be observed that there remains a gap to the fcCEO scenario, which is even more pronounced in smaller networks or lower SNRs. This gap is caused by the successive nature of the applied transmission protocol and the fact that there are no distinct phases for exchanging instantaneous side-information and forwarding information to the common receiver. In particular, due to the specific optimization order, the quality of  $\mathbf{s}_{<m}$  differs for different sensors in the network. To be more precise, the amount of information  $\mathbf{s}_{<m}$  has about the relevant variable  $x$  increases gradually for each additional sensor, see Figure 5.5. This means that later sensors can benefit more from exploiting the instantaneous side-information  $\mathbf{s}_{<m}$  than first sensors. As described earlier, the successive broadcasting protocol might suffer from the curse of dimensionality. On the one hand, this can be a problem during the optimization since pmfs like  $p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}, x)$  can become very large. On the other hand, each sensor has to store the mapping  $p(z_m | y_m, \mathbf{s}_{<m})$  to apply the compression during run-time. Since this mapping also depends on the network size, sensors might run into memory issues. Note that this numerical problem is the reason why there is no result for  $|\mathbb{S}_m| = 8$  and a network size of  $M = 6$  in Figure 5.6 and 5.7 since storing a single instance of  $p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}, x)$  requires 2024 GiB<sup>1</sup>.

### 5.3 Sequential Point-To-Point Protocol

Since broadcasting instantaneous side-information suffers from the curse of dimensionality for larger networks, a successive point-to-point transmission protocol is introduced. The system model is given in Figure 5.8. In general, the setup is very similar to the one for broadcasting instantaneous side-information. Each sensor observes a noisy version  $y_m$  of the same relevant signal  $x$ . Now, each sensor  $m$  has to compress its measurement  $y_m$  in order to fulfill the individual rate constraint  $C_m$  when forwarding the measurements to the common receiver. The difference to the broadcast case is that the instantaneous side-information of sensor  $m$  is only forwarded to the direct successor  $m + 1$ , establishing a sequential chain from the first to the last sensor. The dependencies among the random variables are illustrated in Figure 5.9 using a Markov model. It is easy to see that the independence of different  $\mathcal{Z}_m$  given  $\mathcal{X}$  does not hold anymore, as it does for the original

<sup>1</sup>1 GiB = 1024 MiB, 1 MiB = 1024 KiB, 1 KiB = 1024 byte

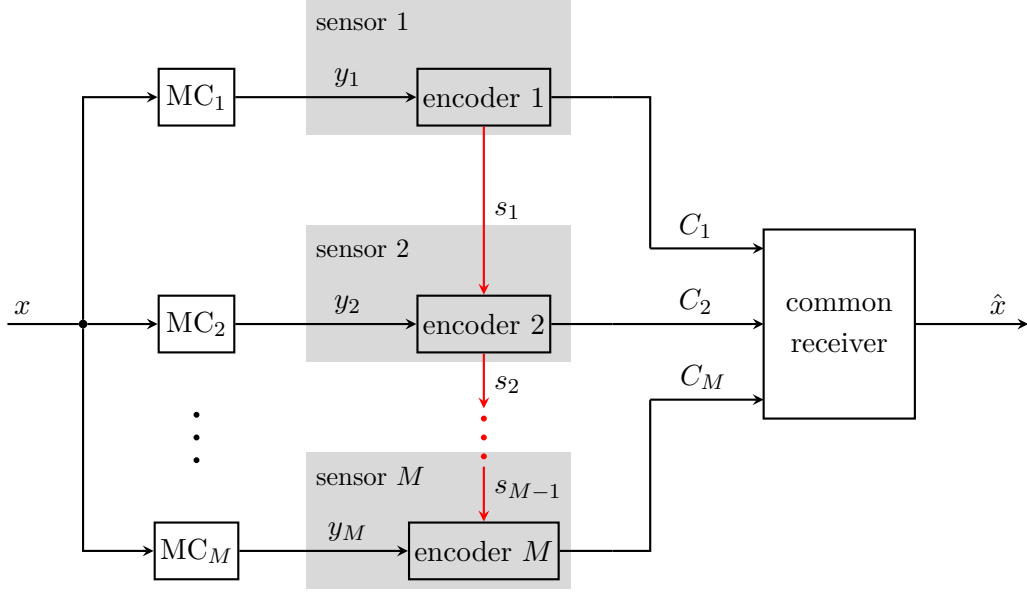


Figure 5.8: System model for the pcCEO scenario using the sequential point-to-point transmission protocol

CEO problem. The different  $\mathcal{Z}_m$  are coupled via the instantaneous side-information  $\mathcal{S}_m$ . This leads to the mathematical structure

$$p(x, \mathbf{y}, \mathbf{z}, \mathbf{s}) = \prod_{m=1}^M p(z_m | y_m, s_{m-1}) p(s_m | y_m, s_{m-1}) p(y_m | x) p(x). \quad (5.10)$$

The encoding process of a specific sensor  $m$  is depicted in Figure 5.10. Each sensor exploits the received instantaneous side-information  $s_{m-1}$ . The generation of  $z_m$  to be forwarded to the common receiver and the output  $s_m$  to be forwarded to the subsequent sensor are generally independent operations. As both outputs shall be as informative as possible about the relevant signal  $x$ , the quantizer mappings  $p(s_m | y_m, s_{m-1}) \in \{0, 1\}$  and  $p(z_m | y_m, s_{m-1}) \in [0, 1]$  can be designed using the IB principle. Figure 5.11 illustrates this by means of IB graphs. Both quantizers fuse the two inputs  $y_m$  and  $s_{m-1}$  to a single output while preserving as much information as possible about the relevant signal  $x$ . Since the mapping  $p(z_m | y_m, s_{m-1})$  is of a stochastic nature, a binary encoding and rate-matching step is required in order to forward a compressed version of  $z_m$  to the common receiver. This further encoding is denoted as BERM in Figure 5.10.

### 5.3.1 Generation of Point-To-Point Side-Information

Similar to the broadcast case, the design of  $p(s_m | y_m, s_{m-1})$  is inspired by the general GDIB algorithm. The instantaneous side-information of the previous sensor  $s_{m-1}$  can be exploited to improve the information  $s_m$  has about  $x$ . The optimization approach to gener-



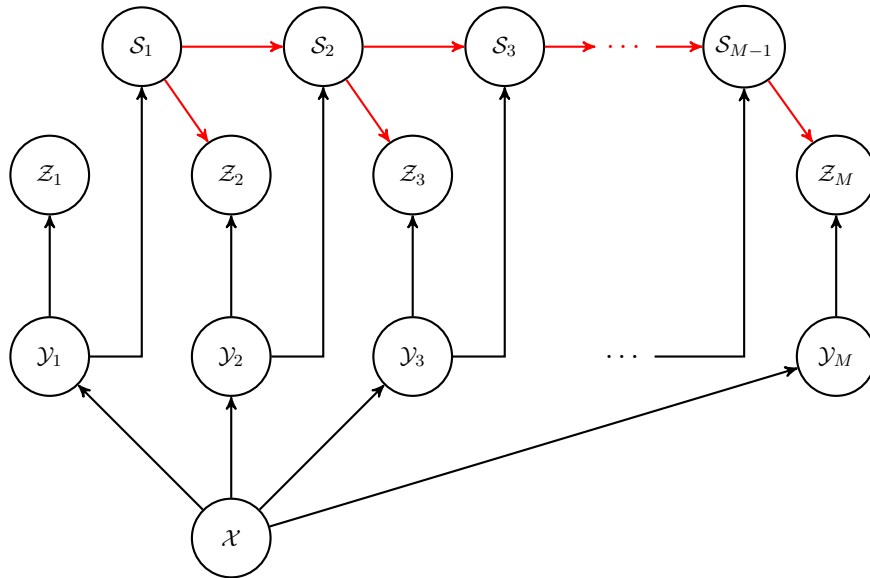


Figure 5.9: Markov model of statistical dependencies for the sequential point-to-point transmission protocol

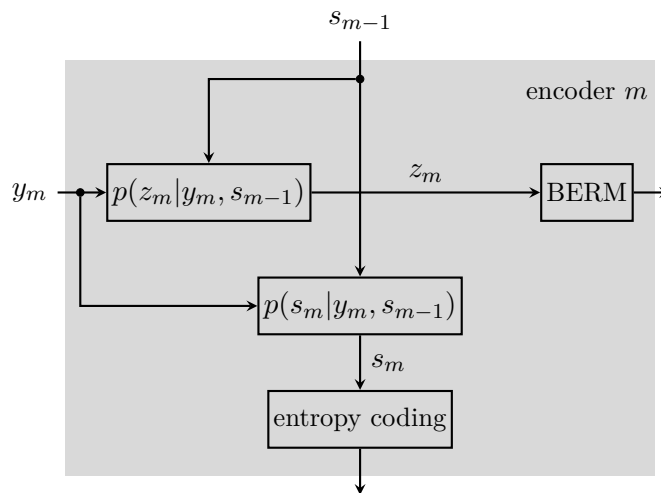


Figure 5.10: Encoding process for the sequential point-to-point transmission protocol

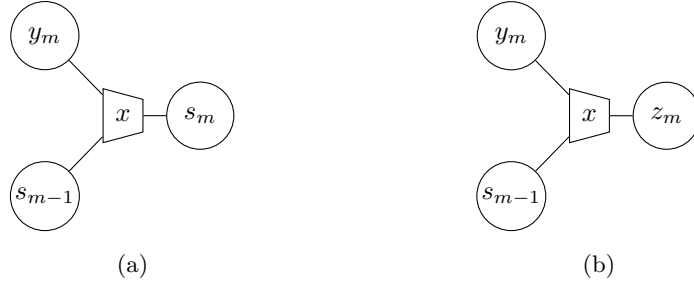


Figure 5.11: Graphical illustration of IB fusion of two inputs to determine instantaneous side-information  $s_m$  (a) and the quantizer sensor output  $z_m$  (b) for the sequential point-to-point transmission protocol

ate the instantaneous side-information applying the sequential point-to-point transmission protocol can be defined as

$$\begin{aligned} L_{\text{PTP-SIDE}}^{(1)} &= I(\mathcal{X}; \mathcal{S}_1) - \beta I(\mathcal{Y}_1; \mathcal{S}_1) \\ &\vdots \\ L_{\text{PTP-SIDE}}^{(M-1)} &= I(\mathcal{X}; \mathcal{S}_{M-1}) - \beta I(\mathcal{Y}_{M-1}, \mathcal{S}_{M-2}; \mathcal{S}_{M-1}). \end{aligned}$$

The first sensor only applies the standard scalar IB method. Subsequent sensors use the instantaneous side-information of the previous sensor  $s_{m-1}$  as an additional input for the IB method. In this way, the instantaneous side-information  $s_m$  increases its information about the relevant variable with each further sensor. In contrast to the broadcast case, the relevant mutual information is not conditioned on  $\mathcal{S}_{<m}$  because sensor  $m$  has only access to  $s_{m-1}$  and not to indices of any other sensor. Therefore, Wyner-Ziv coding cannot be applied here. For sensor  $m$ , the optimization problem becomes

$$L_{\text{PTP-SIDE}}^{(m)}[p(s_m|y_m, s_{m-1})] = I(\mathcal{X}; \mathcal{S}_m) - \beta_m I(\mathcal{Y}_m, \mathcal{S}_{m-1}; \mathcal{S}_m). \quad (5.11)$$

### Algorithmic Solution

The optimization problem given in (5.11) can be solved by taking the derivative w.r.t. the mapping  $p(s_m|y_m, s_{m-1})$  and equating it to zero. Similar to the general IB method, this results in an implicit update equation for sensor  $m$

$$p(s_m|y_m, s_{m-1}) = \frac{e^{-d_{\beta_m}(y_m, s_m, s_{m-1})}}{\sum_{s_m} e^{-d_{\beta_m}(y_m, s_m, s_{m-1})}} \quad (5.12)$$

with

$$d_{\beta_m}(y_m, s_m, s_{m-1}) := \frac{1}{\beta_m} \cdot D_{\text{KL}}[p(x|y_m, s_{m-1}) || p(x|s_m)] - \log p(s_m). \quad (5.13)$$

Since  $p(x|s_m)$  and  $p(s_m)$  are both dependent on the mapping  $p(s_m|y_m, s_{m-1})$ , equation (5.12) is an implicit update equation, which can be solved by an extension of the Blahut-Arimoto algorithm. A detailed derivation of this solution is given in Appendix E.

---

**Algorithm 13:** Extended Blahut-Arimoto algorithm to obtain side-information for the sequential point-to-point transmission protocol

---

```

input      :  $m, p^{\text{init}}(s_m|y_m, s_{m-1}), p(y_m, x), p(s_{m-1}|x), \epsilon$ 
output    :  $p(s_m|y_m, s_{m-1}) \in 0, 1$ 
1 begin
    initialization:
         $p(s_m|y_m, s_{m-1})^{(0)} \leftarrow p^{\text{init}}(s_m|y_m, s_{m-1}),$ 
         $l \leftarrow 1$ 
2    $p(s_{m-1}, y_m, x) = p(s_{m-1}|x)p(y_m|x)p(x)$ 
3    $p(x|s_{m-1}, y_m) = p(s_{m-1}, y_m, x) / \sum_x p(s_{m-1}, y_m, x)$ 
4   do
5       // KL-Divergence  $D_{\text{KL}}(y_m, s_{m-1}, s_m)$  of (5.13)
6        $p(s_m, x)^{(l)} = \sum_{s_{m-1}} \sum_{y_m} p(s_m|y_m, s_{m-1})^{(l-1)} p(s_{m-1}, y_m, x)$ 
7        $p(x|s_m)^{(l)} = p(s_m, x)^{(l)} / \sum_x p(s_m, x)^{(l)}$ 
8        $D_{\text{KL}}(y_m, s_{m-1}, s_m)^{(l)} = \sum_x p(x|s_{m-1}, y_m) \cdot \log \frac{p(x|s_{m-1}, y_m)}{p(x|s_m)^{(l)}}$ 
9       // find minimum of  $D_{\text{KL}}(y_m, s_{m-1}, s_m)$  for all samples  $y_m \in \mathbb{Y}_m$  and
           $s_{m-1} \in \mathbb{S}_{m-1}$ 
10      for  $y_i \in \mathbb{Y}_m$  do
11          for  $s_j \in \mathbb{S}_{m-1}$  do
12               $s_m^*(y_i, s_j)^{(l)} = \arg \min_{s_m} D_{\text{KL}}(y_m, s_{m-1}, s_m)^{(l)}$ 
13              // update quantizer for specific  $y_i$  and  $s_j$ 
14               $p(s_m^*(y_i, s_j)^{(l)}|y_i, s_j)^{(l)} = 1$ 
15           $l \leftarrow l + 1$ 
16      while  $D_{\text{JS}}[p^{(l)}(s_m|y_m, s_{m-1}) || p^{(l-1)}(s_m|y_m, s_{m-1})] > \epsilon$ 

```

---

Similar to the broadcast case, the mapping  $p(s_m|y_m, s_{m-1})$  is supposed to be deterministic using  $\beta_m = 0$ . Therefore, the above update equation simplifies to

$$s_m^*(y_m, s_{m-1}) = \arg \min_{s_m} D_{\text{KL}} [p(x|y_m, s_{m-1}) || p(x|s_m)] \quad (5.14)$$

with the update equation

$$p(s_m|y_m, s_{m-1}) = \begin{cases} 1 & \text{for } s_m(y_m, s_{m-1}) = s_m^*(y_m, s_{m-1}) \\ 0 & \text{else.} \end{cases} \quad (5.15)$$

The extended Blahut-Arimoto algorithm is given in Algorithm 13. This algorithm is very similar to the one for obtaining broadcast side-information. The major difference lies in the available instantaneous side-information  $s_{m-1}$  instead of  $\mathbf{s}_{<m}$ , which leads to slightly different pmfs. However, this difference leads to a significant reduction in memory requirements during the optimization.

**Evolution of Side-Information Available at Sensors:** Figure 5.12 illustrates the evolution of the instantaneous side-information running from sensor to sensor. Therefore,

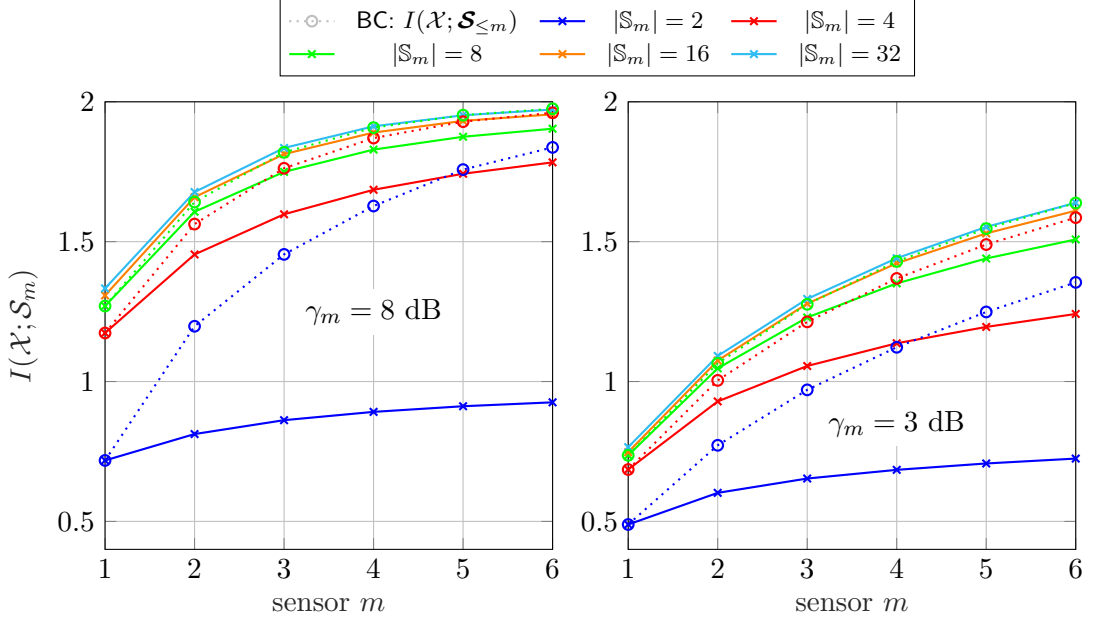


Figure 5.12: Evolution of side-information  $I(\mathcal{X}; \mathcal{S}_m)$  for sensor  $m$  in a network with  $M = 6$  sensors for different cardinalities  $|\mathcal{S}_m|$  for the sequential point-to-point transmission protocol;  $|\mathcal{X}| = 4$ ,  $|\mathcal{Y}_m| = 64$

the solid lines illustrate the relevant mutual side-information  $I(\mathcal{X}; \mathcal{S}_m)$  at a specific sensor  $m$  in a network of size  $M = 6$  applying the sequential point-to-point protocol for different cardinalities  $|\mathcal{S}_m|$ . It can be observed that in both SNRs  $\gamma_m = \{8, 3\}$ , the relevant mutual information  $I(\mathcal{X}; \mathcal{S}_m)$  increases while passing through the network. This shows the effect that the instantaneous side-information  $s_m$  becomes more and more informative about the relevant signal  $x$  for each additional sensor. Naturally, for larger cardinalities  $|\mathcal{S}_m|$ , more information about the relevant signal  $x$  can be preserved, leading to larger  $I(\mathcal{X}; \mathcal{S}_m)$ . The main difference to the broadcast case is that the instantaneous side-information provided to sensor  $m$  is represented by a single highly compressed index  $s_{m-1}$  with cardinality  $|\mathcal{S}_{m-1}|$ . While the resolution  $|\mathcal{S}_{<m}|$  of the available instantaneous side-information  $\mathbf{s}_{<m}$  increases with  $m$  for the successive broadcasting transmission protocol, it remains the same for the sequential point-to-point protocol. Accordingly, a larger cardinality  $|\mathcal{S}_m|$  is required to not introduce additional compression losses. For comparison, the dotted lines represent the results of the broadcast case given in Figure 5.5. In particular, they illustrate  $I(\mathcal{X}; \mathcal{S}_{\leq m})$  incorporating the instantaneous side-information  $\mathbf{s}_{<m}$  of all previous sensors. It can be observed that the broadcast case gains much more relevant mutual side-information than the sequential point-to-point transmission. However, by choosing the cardinality  $|\mathcal{S}_m|$  large enough, the difference between the broadcast and the point-to-point transmission becomes negligible.

### 5.3.2 Generate Information Forwarded to the Common Receiver

Extending the GDIB approach for the pcCEO scenario using the sequential point-to-point transmission protocol defines the Greedy Distributed Information Bottleneck-Point-

to-Point Side-Information (GDIB-PTP) approach. By incorporating the instantaneous side-information  $s_m$ , the optimization approach can be defined as

$$\begin{aligned} L_{\text{GDIB-PTP}}^{(1)} &= I(\mathcal{X}; \mathcal{Z}_1) - \beta_1 I(\mathcal{Y}_1; \mathcal{Z}_1) \\ &\vdots \\ L_{\text{GDIB-PTP}}^{(M)} &= I(\mathcal{X}; \mathcal{Z}_M | \mathcal{Z}_{<M}) - \beta_M I(\mathcal{Y}_M, \mathcal{S}_{M-1}; \mathcal{Z}_M | \mathcal{Z}_{<M}). \end{aligned}$$

The main difference to the original GDIB optimization of Subsection 4.2 lies in the compression rate  $I(\mathcal{Y}_m, \mathcal{S}_{m-1}; \mathcal{Z}_m | \mathcal{Z}_{<m})$  which emerges from the combination of the instantaneous side-information of the previous sensor  $s_{m-1}$  and the observation  $y_m$  of sensor  $m$ . It becomes obvious that the first sensor is optimized by a scalar IB approach, exploiting no side-information at all. Subsequent sensors are optimized applying Wyner-Ziv coding similar to the original GDIB optimization, as well as the instantaneous side-information  $s_{m-1}$  from the previous sensor. Hence, for sensor  $m$ , the optimization problem becomes

$$L_{\text{GDIB-PTP}}^{(m)}[p(z_m | y_m, s_{m-1})] = I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{<m}) - \beta_m I(\mathcal{Y}_m, \mathcal{S}_{m-1}; \mathcal{Z}_m | \mathcal{Z}_{<m}). \quad (5.16)$$

### Algorithmic Solution

The optimization problem given in (5.16) can be solved by taking the derivative w.r.t. the mapping  $p(z_m | y_m, s_{m-1})$  and equating it to zero. This results in the implicit update equation for sensor  $m$

$$p(z_m | y_m, s_{m-1}) = \frac{e^{-d_{\beta_m}(y_m, z_m, s_{m-1})}}{\sum_{z_m} e^{-d_{\beta_m}(y_m, z_m, s_{m-1})}} \quad (5.17)$$

with

$$\begin{aligned} d_{\beta_m}(y_m, z_m, s_{m-1}) &:= \mathbb{E}_{\mathcal{Z}_{<m} | y_m, s_{m-1}} \left[ \frac{1}{\beta_m} \right. \\ &\quad \left. D_{\text{KL}} [p(x | y_m, s_{m-1}, \mathbf{z}_{<m}) || p(x | \mathbf{z}_{\leq m})] - \log p(z_m | \mathbf{z}_{<m}) \right]. \end{aligned} \quad (5.18)$$

Since  $p(x | \mathbf{z}_{\leq m})$  and  $p(z_m | \mathbf{z}_{<m})$  depend on the mapping  $p(z_m | y_m, s_{m-1})$ , (5.17) is an implicit equation. Using an extension of the Blahut-Arimoto algorithm, this update equation can be solved, resulting in local optimal solutions. A detailed derivation of the above solution and the involved pmfs is given in Appendix F.

The extended Blahut-Arimoto algorithm solving the given optimization problem for sensor  $m$  and a specific  $\beta_m$  is given in Algorithm 14. The instantaneous side-information is given as  $p(s_i | y_i, s_{i-1}) \forall i \leq m$ . Moreover, for the optimization, the mapping  $p(y_i, x) \forall i \leq m$  is not only required for the current sensor  $m$ , but for all previously designed quantizers. As explained in Appendix F the required pmfs for this algorithm can be computed recursively. Therefore, when optimizing sensor  $m$ , these pmfs are considered as input variables. In particular,  $p(z_i | y_i, s_{i-1})$ ,  $p(s_i | \mathbf{z}_{\leq i}, x) \forall i < m$  and  $p(\mathbf{z}_{<m-1}, x)$  are already computed during

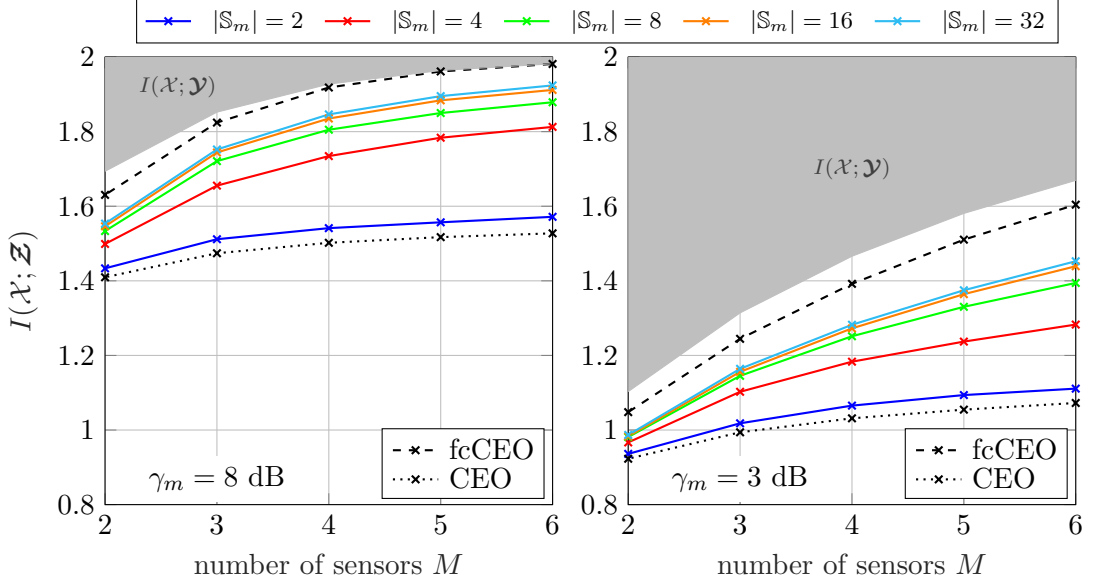


Figure 5.13: Relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  versus the network size for a fixed sum-rate of  $C_{\text{sum}} = 2.5$  bit/s/Hz and  $C_m = \frac{C_{\text{sum}}}{M}$  for the sequential point-to-point transmission protocol with different cardinalities  $|\mathcal{S}_m|$ ;  $|\mathcal{X}| = 4$ ,  $|\mathcal{Y}_m| = 64$ ,  $|\mathcal{Z}_m| = 4$

the optimization of previous sensors and are used as input variables for optimizing the current sensor  $m$ .

Similar to the broadcast case, the flowchart, which describes the original GDIB algorithm in Figure 4.1 is also valid for partially cooperating sensors using the sequential point-to-point transmission protocol. The difference is mainly the extended Blahut-Arimoto algorithm. Moreover, the compression rate used to find the rate fulfilling parameter  $\beta_m$  in the bisection search is calculated as  $I(\mathcal{Y}_m, \mathcal{S}_{m-1}; \mathcal{Z}_m | \mathcal{Z}_{<m})$ , as given in (5.16). As already pointed out, solutions might not be optimal since the algorithm does not necessarily find extreme points of the solution space.

Note that in contrast to the broadcast case, the mapping and the required pmfs used within the optimization are only dependent on the instantaneous side-information of the previous sensor. This reduces the required memory complexity significantly. Moreover, as described in Section 4.3, the dependency on  $\mathbf{z}_{<m}$  can also be relaxed by compressing all  $\mathbf{z}_{<m}$  to a scalar  $z_{<m}^*$  removing the dependency on the network size completely.

**Performance Analysis for different Network Sizes:** Figure 5.13 and 5.14 illustrate the overall performance of the pcCEO scenario applying the sequential point-to-point transmission protocol. Therefore, the same symmetric scenario as for the broadcast case is considered, where all sensors share the same channel in an orthogonal way and in a round-robin fashion. The forward channel to the common receiver has a fixed sum-rate  $C_{\text{sum}} = \sum_{m=1}^M C_m$  which is equally distributed on each sensor  $C_m = \frac{C_{\text{sum}}}{M}$ . The relevant signal is chosen to be a 4-ASK signal. The upper and lower bound are depicted in black, representing a fcCEO scenario and a non-cooperative CEO scenario. In general, the curves are very similar to those for broadcasting instantaneous side-information in Figure 5.6 and 5.7. It can be observed that the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  increases

---

**Algorithm 14:** Extended Blahut-Arimoto algorithm for the sequential point-to-point transmission protocol

---

**input** :  $m, p^{\text{init}}(z_m|y_m, s_{m-1}), p(y_i, x), p(s_i|y_i, s_{i-1}) \forall i \leq m, \beta_m, \epsilon$

recursively calculated inputs from previous sensor optimizations:  
 $p(z_i|y_i, s_{i-1}), p(s_i|\mathbf{z}_{\leq i}, x) \forall i < m, p(\mathbf{z}_{< m-1}, x)$

**output** :  $p(z_m|y_m, s_{m-1}) \in [0, 1], p(\mathbf{z}_{< m}, x), p(s_m|\mathbf{z}_{\leq m}, x)$

**1 begin**

**initialization:**

$p(z_m|y_m, s_{m-1})^{(0)} \leftarrow p^{\text{init}}(z_m|y_m, s_{m-1}),$   
 $l \leftarrow 1$

**2**  $p(z_{m-1}, s_{m-1}|x, \mathbf{z}_{< m-1}) = \sum_{s_{m-2}} p(s_{m-2}|\mathbf{z}_{\leq m-2}, x) \cdot$   
 $\sum_{y_{m-1}} p(z_{m-1}|y_{m-1}, s_{m-2})p(s_{m-1}|y_{m-1}, s_{m-2})p(y_{m-1}|x)$

**3**  $p(\mathbf{z}_{< m-1}, y_m, x) = p(\mathbf{z}_{< m-1}, x)p(y_m|x)$

**4**  $p(y_m, s_{m-1}, \mathbf{z}_{< m}, x) = p(z_{m-1}, s_{m-1}|x, \mathbf{z}_{< m-1})p(\mathbf{z}_{< m-1}, y_m, x)$

**5**  $p(x|y_m, s_{m-1}, \mathbf{z}_{< m}) = p(y_m, s_{m-1}, \mathbf{z}_{< m}, x) / \sum_x p(y_m, s_{m-1}, \mathbf{z}_{< m}, x)$

**6**  $p(\mathbf{z}_{< m}|y_m, s_{m-1}) = \sum_x p(y_m, s_{m-1}, \mathbf{z}_{< m}, x) \sum_x \sum_{\mathbf{z}_{< m}} p(y_m, s_{m-1}, \mathbf{z}_{< m}, x)$

**7 do**

**8** // KL-Divergence  $D_{\text{KL}}(y_m, s_{m-1}, \mathbf{z}_{\leq m})$  of (5.18)

**9**  $p(\mathbf{z}_{\leq m}, x)^{(l)} = \sum_{s_{m-1}} p(z_{m-1}, s_{m-1}|x, \mathbf{z}_{< m-1}) \cdot$   
 $\sum_{y_m} p(z_m|y_m, s_{m-1})^{(l-1)}p(\mathbf{z}_{< m-1}, y_m, x)$

**10**  $p(x|\mathbf{z}_{\leq m})^{(l)} = p(\mathbf{z}_{\leq m}, x)^{(l)} / \sum_x p(\mathbf{z}_{\leq m}, x)^{(l)}$

**11**  $D_{\text{KL}}(y_m, s_{m-1}, \mathbf{z}_{\leq m})^{(l)} = \sum_x p(x|y_m, s_{m-1}, \mathbf{z}_{< m}) \cdot \log \frac{p(x|y_m, s_{m-1}, \mathbf{z}_{< m})}{p(x|\mathbf{z}_{\leq m})^{(l)}}$

**12** // distance  $d_{\beta_m}(y_m, z_m, s_{m-1})$  (5.18)

**13**  $p(\mathbf{z}_{\leq m})^{(l)} = \sum_x p(\mathbf{z}_{\leq m}, x)^{(l)}$

**14**  $p(z_m|\mathbf{z}_{< m})^{(l)} = p(\mathbf{z}_{\leq m})^{(l)} / \sum_{z_m} p(\mathbf{z}_{\leq m})^{(l)}$

**15**  $d_{\beta_m}(z_m, y_m, s_{m-1})^{(l)} =$   
 $\sum_{\mathbf{z}_{< m}} p(\mathbf{z}_{< m}|y_m, s_{m-1}) \cdot \left[ \frac{1}{\beta_m} D_{\text{KL}}(y_m, s_{m-1}, \mathbf{z}_{\leq m})^{(l)} - \log p(z_m|\mathbf{z}_{< m})^{(l)} \right]$

**16** // update quantizer  $p(z_m|y_m, s_{m-1})$

**17**  $p(z_m|y_m, s_{m-1})^{(l)} = \frac{1}{\sum_z e^{-d_{\beta_m}(y_m, z_m, s_{m-1})^{(l)}}} e^{-d_{\beta_m}(y_m, z_m, s_{m-1})^{(l)}}$

**18**  $l \leftarrow l + 1$

**19 while**  $D_{\text{JS}}[p^{(l)}(z_m|y_m, s_{m-1}) || p^{(l-1)}(z_m|y_m, s_{m-1})] > \epsilon$

**20** // calculate  $p(s_m|\mathbf{z}_{\leq m}, x)$  and  $p(\mathbf{z}_{< m}, x)$  for recursive input of successive sensor

**21**  $p(\mathbf{z}_{< m}, x) = \sum_{z_m} p(\mathbf{z}_{\leq m}, x)$

**22**  $p(y_m, x, z_m) = p(y_m, x) \sum_{s_{m-1}} p(z_m|y_m, s_{m-1})p(s_{m-1}|x)$

**23**  $p(y_m|x, z_m) = p(y_m, x, z_m) / \sum_{y_m} p(y_m, x, z_m)$

**24**  $p(s_m|\mathbf{z}_{\leq m}, x) = \sum_{s_{m-1}} p(s_{m-1}|\mathbf{z}_{\leq m-1}, x) \sum_{y_m} p(y_m|x, z_m)p(s_m|y_m, s_{m-1})$

---

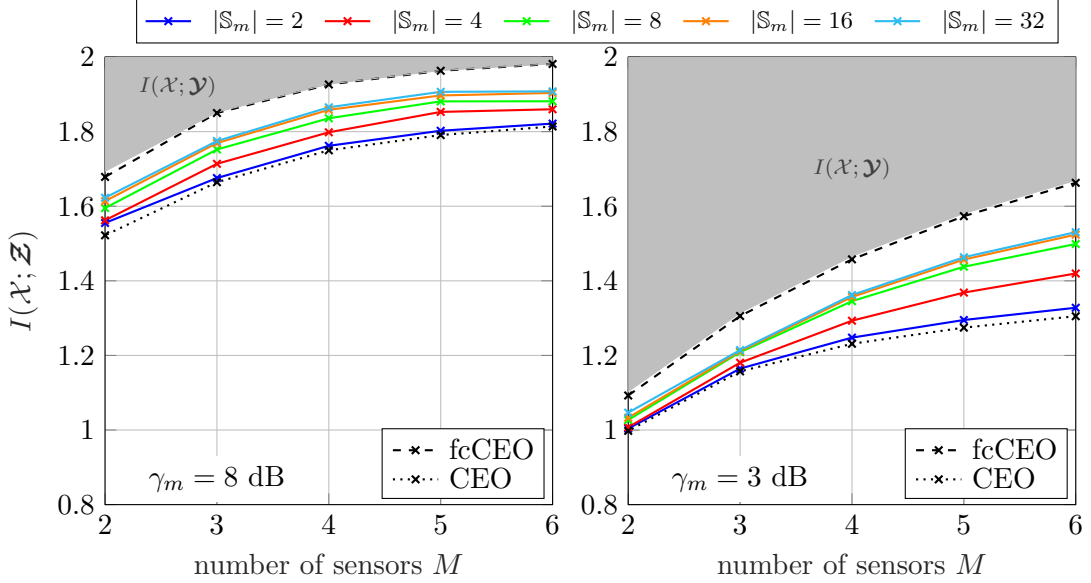


Figure 5.14: Relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  versus the network size for a fixed sum-rate of  $C_{\text{sum}} = 4$  bit/s/Hz and  $C_m = \frac{C_{\text{sum}}}{M}$  for the sequential point-to-point transmission protocol with different cardinalities  $|\mathcal{S}_m|$ ;  $|\mathcal{X}| = 4$ ,  $|\mathcal{Y}_m| = 64$ ,  $|\mathcal{Z}_m| = 4$

for larger networks. This is independent of the SNR or the sum-rate  $C_{\text{sum}}$  and still occurs without exchanging instantaneous side-information. However, exchanging instantaneous side-information increases the performance compared to the non-cooperative case, even if just a single bit is exchanged, i.e.,  $|\mathcal{S}_m| = 2$ . However, the instantaneous side-information with cardinality  $|\mathcal{S}_m| = 2$  can improve the performance just a little bit since small cardinalities  $|\mathcal{S}_m|$  requires a strong compression with massive losses. Although  $\mathcal{S}_m$  gets more reliable for each additional sensor in the network, the information it contains about  $\mathcal{X}$  saturates early due to the accumulated loss of multiple compression steps. Incrementing the cardinality  $|\mathcal{S}_m|$  leads to significant gains compared to the non-cooperative CEO scenario. However, similar to the broadcast case, there still remains a gap to the fcCEO scenario even for large  $|\mathcal{S}_m|$ . Again, this can be explained by the successive communication strategy, since sensors at the beginning of the optimization chain can only exploit no or little instantaneous side-information, and the fact that cooperation with other sensors and forwarding to the common receiver is not separated.

**Performance Analysis for different Sum-Rates:** Figure 5.15 analyzes the performance of the pcCEO scenario applying the sequential point-to-point transmission protocol for different sum-rates. Therefore, the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  is depicted versus different sum-rates  $C_{\text{sum}} = \sum_{m=1}^M C_m$  for a symmetric scenario with  $M = 5$  sensors. The sum-rate is equally distributed on each forward link to the common receiver, i.e.,  $C_m = \frac{C_{\text{sum}}}{M}$  which is why larger sum-rates correlate with higher individual link capacities. The relevant signal is chosen to be a 4-ASK signal. Again, the non-cooperative CEO scenario serves as a lower bound, while the fcCEO scenario serves as an upper bound. In general, increasing the sum-rate leads to a larger overall performance up to a certain level. It can be observed that a cardinality of  $|\mathcal{S}_m| = 2$  only leads to a small gain in



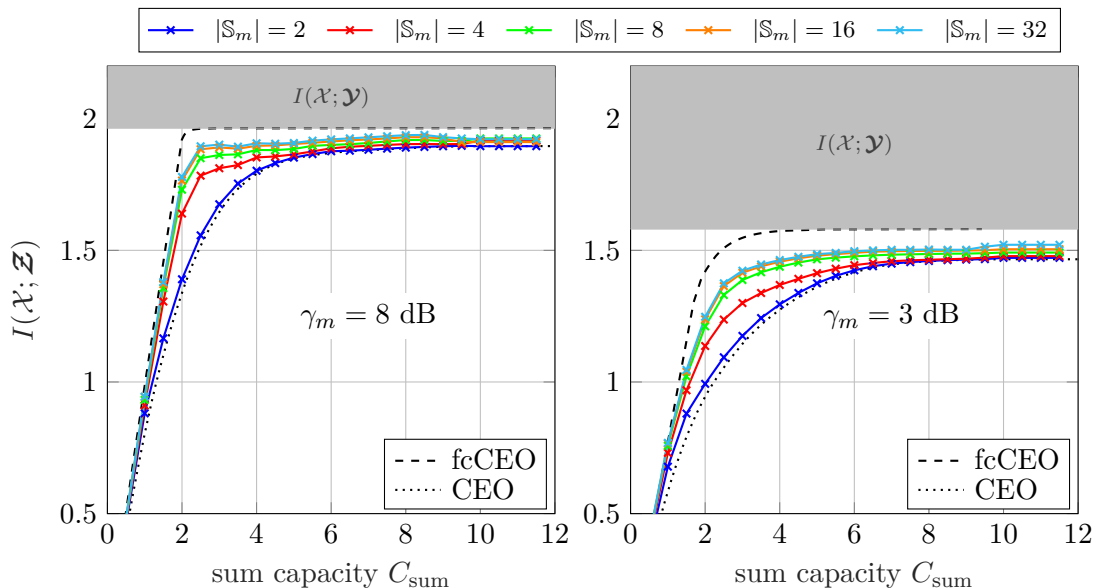


Figure 5.15: Relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  versus sum-rate  $C_{\text{sum}}$  with  $C_m = \frac{C_{\text{sum}}}{M}$  using the sequential point-to-point transmission protocol with different cardinalities  $|\mathcal{S}_m|$ ;  $|\mathcal{X}| = 4$ ,  $|\mathcal{Y}_m| = 64$ ,  $|\mathcal{Z}_m| = 4$

performance. This can be explained by the incremental compression loss when representing the instantaneous side-information with only one bit. Increasing the cardinality  $|\mathcal{S}_m|$ , the improvement compared to the non-cooperative CEO scenario gets more and more significant. In general, the gain of cooperation is largest for lower sum-rates. Comparing the results to the upper bound illuminates the loss due to limited available instantaneous side-information at early transmitting sensors.

**Performance Analysis in an Asymmetric Scenario:** The analysis of the original GDIB algorithm in asymmetric non-cooperative CEO scenarios revealed that this algorithm is very sensitive to the optimization order, i.e., the Wyner-Ziv coding strategy. The question arises if partial cooperation among sensors can improve the robustness against bad Wyner-Ziv coding strategies. In order to compare the results to the non-cooperative CEO scenario, the same two asymmetric setups are investigated as in Figure 4.12. Scenario 1 considers the case where sensors with bad measurement SNRs  $\gamma_m$  have low link capacities  $C_m$  while sensors with good measurement SNRs  $\gamma_m$  have high link capacities  $C_m$ . Scenario 2 considers the opposite case where sensors with bad SNRs have high link capacities and vice versa. The relevant signal  $x$  is chosen to be a 4-ASK signal.

Figure 5.16 illustrates the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  for all  $M!$  sensor permutations, i.e., all different Wyner-Ziv coding strategies for a network of  $M = 4$  sensors. The dots show the results of the non-cooperative CEO scenario as already discussed in Figure 4.12. The results for the pcCEO scenario using the sequential point-to-point transmission protocol are depicted as bars. Scenario 1 is depicted in blue, whereas Scenario 2 is depicted in red.

Comparing the overall performance of the two different approaches in general, it is obvious that the relevant information  $I(\mathcal{X}; \mathcal{Z})$  is larger in the pcCEO scenario compared to the

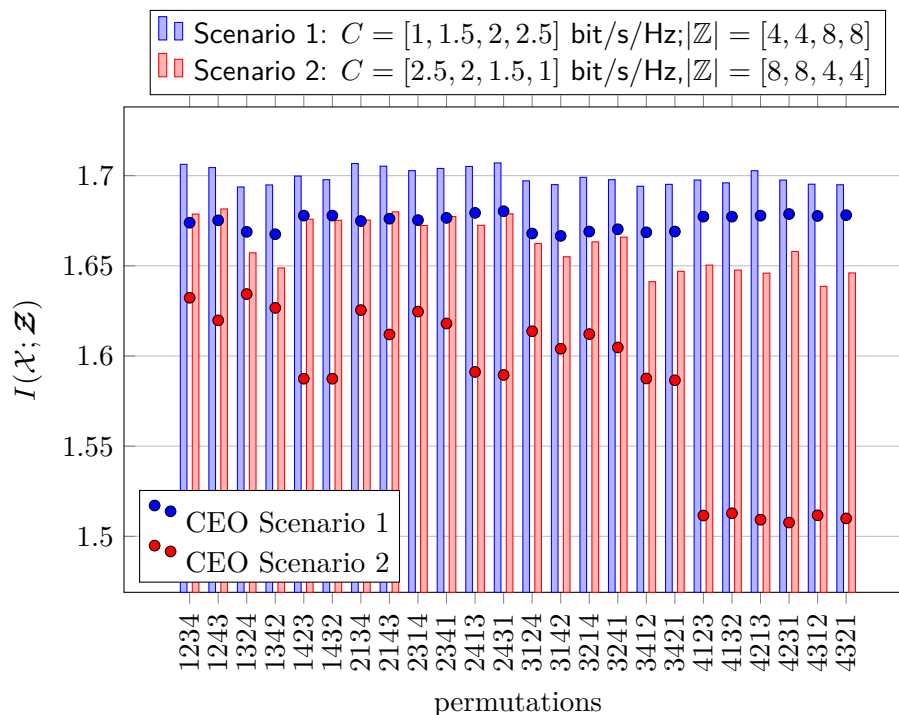


Figure 5.16: Relevant mutual information for an asymmetric scenario with  $M = 4$  sensors, SNRs  $\gamma_m = [2, 4, 6, 8]$  dB and  $|\mathcal{X}| = 4$ ,  $|\mathcal{Y}_m| = 64$ ,  $|\mathcal{Z}_m| = 4$  using the sequential point-to-point transmission protocol with  $|\mathcal{S}_m| = 8$

non-cooperative CEO scenario. Indeed, this can be explained by each sensor exploiting the instantaneous side-information of the previous sensor. Regarding Scenario 1, only minor differences in the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  can be observed between the different optimization orders. Hence, in the case where good sensors are paired with good measurement SNRs, the Wyner-Ziv coding strategy does not have a big impact in the pcCEO scenario as well as in the non-cooperative CEO scenario.

In general, the performance for Scenario 2 is worse than for Scenario 1, again for both the non-cooperative CEO scenario and the pcCEO scenario. Here, accurate measurements have to be strongly compressed in order to be able to forward them to the common receiver. In contrast, unreliable measurements can not contribute much to the overall performance, although they can be forwarded to the common receiver at high rates. It becomes very apparent that the performance difference between different optimization orders is much smaller in the pcCEO scenario compared to the non-cooperative CEO scenario. This can be explained by sensors being able to forward their good measurements to the neighboring sensor, although they might have a bad forward channel to the common receiver. The following sensor can then exploit the instantaneous side-information created by these good measurements. Therefore, exchanging instantaneous side-information can improve the robustness against bad optimization orders.

**Influence of the Relevant Signal:** Figure 5.17 illustrates the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  versus the network size in a symmetric scenario for a Gaussian relevant signal sampled with  $|\mathcal{X}| = 64$  equidistant bins. As before, all sensors share the same for-

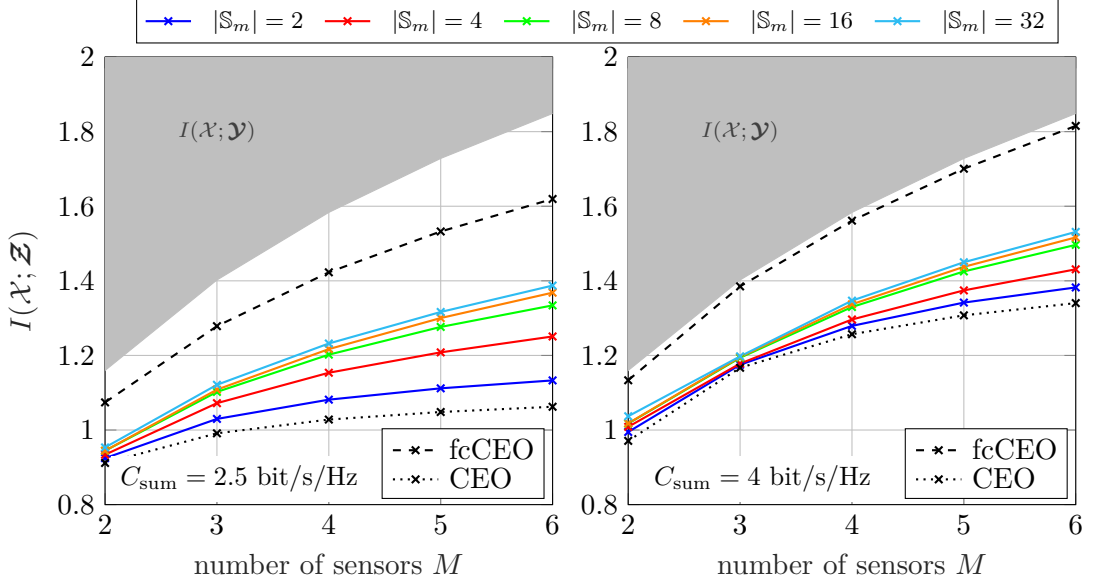


Figure 5.17: Relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  versus the network size for a fixed sum-rate of  $C_{\text{sum}} \in \{2.5, 4\}$  bit/s/Hz,  $\gamma_m = 3$  dB and  $C_m = \frac{C_{\text{sum}}}{M}$  for the sequential point-to-point transmission protocol with different cardinalities  $|\mathcal{S}_m|$  and a Gaussian relevant signal  $x$ ;  $|\mathbb{X}| = 64$ ,  $|\mathbb{Y}_m| = 64$ ,  $|\mathbb{Z}_m| = 4$

ward channel with a sum-rate of  $C_{\text{sum}} \in \{2.5, 4\}$  in an orthogonal way and a round-robin fashion. The measurement SNR is chosen to be  $\gamma_m = 3$  dB. It can be observed that the results are qualitatively the same as for a 4-ASK signal. Generally, instantaneous side-information improves the overall performance. Moreover, representing this instantaneous side-information with more bits results in a larger overall improvement. However, as observed in Figure 5.12, the amount of information a sensor can contribute to  $I(\mathcal{X}; \mathcal{S}_m)$  gets smaller for each additional sensor. Therefore, the gain of larger  $|\mathcal{S}_m|$  shrinks, and there still remains a gap to the fcCEO scenario, even for  $|\mathcal{S}_m| = 32$ .

Figure 5.18 illustrates the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  versus different sum-rates in a symmetric scenario for a Gaussian relevant signal with  $|\mathbb{X}| = 64$  bins. Therefore,  $I(\mathcal{X}; \mathcal{Z})$  is depicted versus different sum-rates  $C_{\text{sum}} = \sum_{m=1}^M C_m$  for a scenario with  $M = 5$  sensors. The signal-to-noise ratio is chosen to be  $\gamma_m = 3$  dB. As before, the sum-rate is equally distributed on each forward link to the common receiver, i.e.,  $C_m = \frac{C_{\text{sum}}}{M}$  which is why larger sum-rates correlate with higher individual link capacities. The general performance is very similar to the performance for a 4-ASK relevant signal. With increasing sum-rates, the overall performance increases, independent of the cardinality  $|\mathcal{S}_m|$ . Similar to previous simulations, a higher cardinality  $|\mathcal{S}_m|$  also results in a larger relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$ . However, there still remains a gap to the fcCEO scenario.

These results indicate that the GDIB-PTP algorithm can also be applied for pre-quantized continuous distributions of the relevant signal.

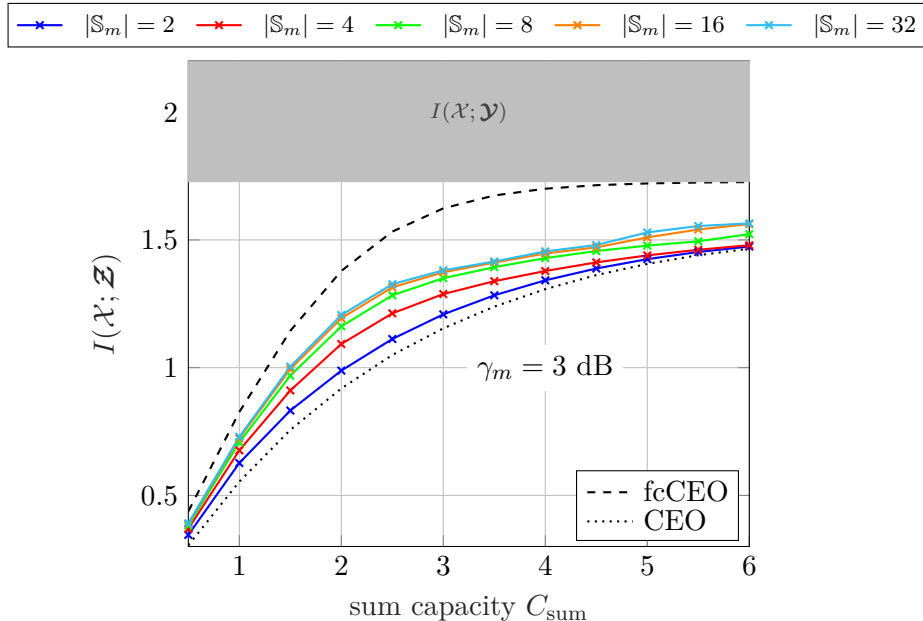


Figure 5.18: Relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  versus sum-rate  $C_{\text{sum}}$  in a network of  $M = 5$  sensors with  $C_m = \frac{C_{\text{sum}}}{M}$  using the sequential point-to-point transmission protocol with different cardinalities  $|\mathcal{S}_m|$  and a Gaussian relevant signal with  $|\mathbb{X}| = 64$ ;  $|\mathbb{Y}_m| = 128$ ,  $|\mathbb{Z}_m| = 4$

## 5.4 Two-Phase Transmission Protocol with Artificial Side-Information

The analysis of the last two transmission protocols revealed that partial cooperation among sensors improves the overall performance compared to the non-cooperative CEO scenario. However, there still remains a gap to the fcCEO scenario. This gap results from the successive exchange of instantaneous side-information and the fact that this instantaneous side-information needs to be quantized in order to be exchanged among sensors. Therefore, the available amount of side-information differs for each sensor in the network. This leads to early sensors having no or little instantaneous side-information to exploit. In order to overcome this, a third transmission protocol consisting of two distinct phases is considered, including the cooperation phase and the transmission phase. Inspired by the fcCEO scenario, the first cooperation phase is used to exchange instantaneous side-information between all sensors until each sensor has approximately the same amount of side-information. The second transmission phase is used to forward the information to the common receiver in the usual way. The difference to the fcCEO scenario is that only compressed versions of the observations can be exchanged during the cooperation phase. For simplicity, it can be assumed that each sensor obtains the same instantaneous side-information represented by  $s^*$ . This shall hold independent of its position in the optimization chain.<sup>2</sup> Within the following investigations, the extrinsic information transfer (EXIT) chart philosophy [Bri01] can be pursued, where extrinsic information is artificially

<sup>2</sup>Strictly speaking, this only holds for symmetric scenarios. This is why no asymmetric scenarios are considered for this communication protocol.

created to analyze the information exchange between decoders in concatenated coding schemes. In the context of the pcCEO scenario using the two-phase transmission protocol, the artificial side-information can be interpreted as extrinsic information about the relevant signal  $x$ . This extrinsic information can be generated by adding AWGN to  $x$ . In this way, the amount of information can be adapted by varying the noise variance to obtain a specific SNR  $\gamma_{\text{extr}}$ . Equivalently, a desired mutual information  $I(\mathcal{X}; \mathcal{S}^*)$  can be adjusted. In order to obtain general conclusions, it is assumed that  $\gamma_{\text{extr}}$  can be chosen independently from the measurement SNRs at the sensors. Artificially creating the side-information implies an important simplification of the Markovian structure, which is necessary to solve the following optimization problem. To be more specific, since  $s^*$  is created artificially, it is assumed to be independent of the indexes  $y_m$  given the relevant signal  $x$ , i.e.,  $p(y_m, s^*|x) = p(y_m|x)p(s^*|x)$  holds. Therefore, the Markovian structure equals the one of the non-cooperative CEO problem. This indicates that in contrast to the first two communication protocols, the supermodularity holds for the two-phase transmission protocol as it also does for the non-cooperative CEO scenario and the fcCEO scenario. Therefore, a greedy optimization structure is optimal, i.e., it finds the extreme points in the solution space. This model leads to the modified optimization problem

$$L_{\text{GDIB-TP}}^{(1)} = I(\mathcal{X}; \mathcal{Z}_1) - \beta_1 I(\mathcal{Y}_1, \mathcal{S}^*; \mathcal{Z}_1) \quad (5.19a)$$

$$\vdots$$

$$L_{\text{GDIB-TP}}^{(M)} = I(\mathcal{X}; \mathcal{Z}_M | \mathcal{Z}_{<M}) - \beta_M I(\mathcal{Y}_M, \mathcal{S}^*; \mathcal{Z}_M | \mathcal{Z}_{<M}). \quad (5.19b)$$

The optimization problem for sensor  $m$  can be solved using the same strategy as described in Section 5.2 and 5.3 leading to the implicit update equation

$$p(z_m | y_m, s^*) = \frac{e^{-d_{\beta_m}(y_m, z_m, s^*)}}{\sum_{z_m} e^{-d_{\beta_m}(y_m, z_m, s^*)}} \quad (5.20)$$

with

$$d_{\beta_m}(y_m, z_m, s^*) := \mathbb{E}_{\mathcal{Z}_{<m} | y_m, s^*} \left[ \frac{1}{\beta_m} \cdot D_{\text{KL}} [p(x | y_m, s^*, \mathbf{z}_{<m}) || p(x | \mathbf{z}_{\leq m})] - \log p(z_m | \mathbf{z}_{<m}) \right]. \quad (5.21)$$

Equivalent to the other communication protocols, the update equation in (5.20) can be solved using a Blahut-Arimoto-like algorithm.

**Performance of Two-Phase Transmission:** In order to investigate the performance of the two-phase transmission protocol, the previous experiment of Figure 5.6 and Figure 5.13 using the successive broadcasting protocol and the sequential point-to-point transmission protocol will also be investigated applying the two-phase transmission protocol. Therefore, Figure 5.19 illustrates the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  versus the network size for a fixed sum-rate of  $C_{\text{sum}} = 2.5$  bit/s/Hz and  $C_m = \frac{C_{\text{sum}}}{M}$ . The extrinsic

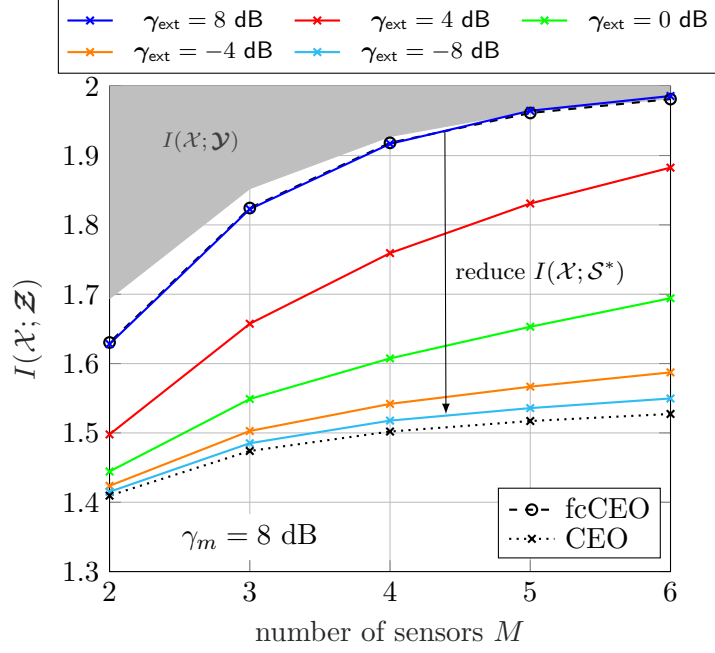


Figure 5.19: Relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  versus the network size for a fixed sum-rate of  $C_{\text{sum}} = 2.5$  bit/s/Hz and  $C_m = \frac{C_{\text{sum}}}{M}$  using a two-phase transmission protocol for artificially decoupled extrinsic information with different  $\gamma_{\text{ext}}$ ;  $\gamma_m = 8$  dB,  $|\mathcal{X}| = 4$ ,  $|\mathcal{Y}_m| = 64$ ,  $|\mathcal{Z}_m| = 4$ ,  $|\mathcal{S}^*| = 512$

information is chosen independent of the measurement SNR. Hence,  $\gamma_{\text{ext}}$  needs not to be the same as  $\gamma_m$ . The cardinality of the extrinsic information is chosen as  $|\mathcal{S}^*| = 512$  to not introduce significant compression losses. The black dashed line represents the upper bound, i.e., the fcCEO scenario. The curve for  $\gamma_{\text{extr}} = \gamma_m = 8$  dB represents the case where each sensor forwards instantaneous side-information with the same quality as its measurement. Therefore,  $\gamma_m$  equals  $\gamma_{\text{extr}}$ . It can be observed that the same performance as for the fcCEO scenario is achieved. This demonstrates that by applying appropriate cooperation among sensors, the remaining gap to the fcCEO scenario can be closed. Decreasing the SNR of the extrinsic information  $\gamma_{\text{ext}}$  or equivalently  $I(\mathcal{X}; \mathcal{S}^*)$  leads to a lower overall performance  $I(\mathcal{X}; \mathcal{Z})$ .

**Influence of Extrinsic Information:** Figure 5.20 illustrates the influence of extrinsic information applying the two-phase transmission protocol. Inspired by EXIT charts, the relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  is depicted versus the extrinsic mutual information  $I(\mathcal{X}; \mathcal{S}^*)$  for different network sizes. Again, all sensors share the same channel in an orthogonal way and a round-robin fashion with a fixed sum-rate of  $C_{\text{sum}} = 2.5$  bit/s/Hz and  $C_m = \frac{C_{\text{sum}}}{M}$ . In the case of  $I(\mathcal{X}; \mathcal{S}^*) = 0$  where no extrinsic information is provided to the sensors, the same performance as in the non-cooperative CEO scenario can be achieved. Increasing the extrinsic information improves the overall relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$ . Since the relevant signal is chosen to be a 4-ASK signal, the overall maximum is 2 bit/s/Hz.

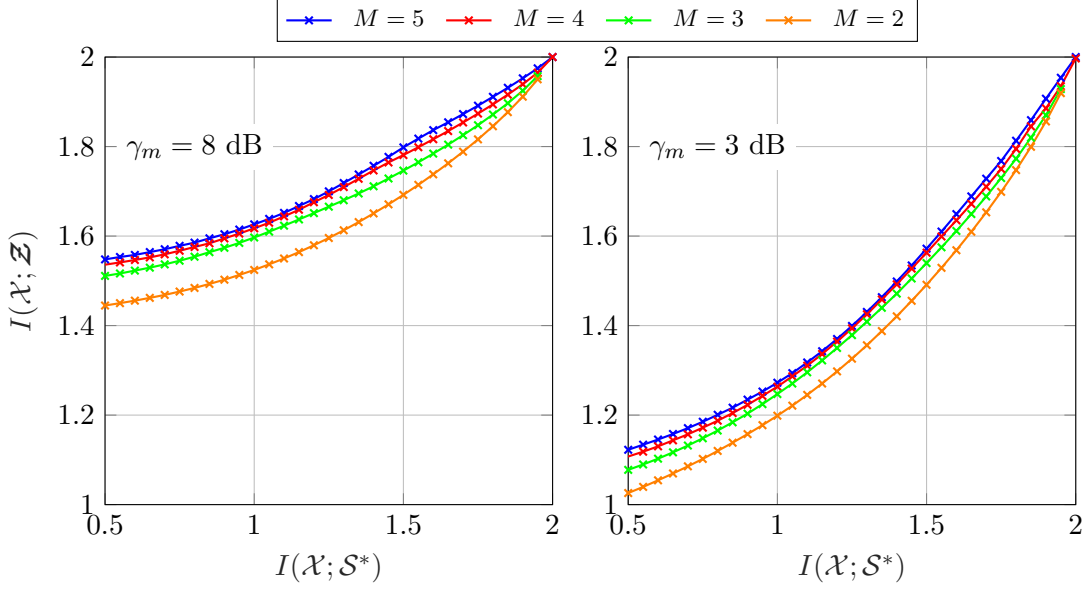


Figure 5.20: Relevant mutual information  $I(\mathcal{X}; \mathcal{Z})$  versus extrinsic mutual information  $I(\mathcal{X}; \mathcal{S}^*)$  for different network sizes and a fixed sum-rate of  $C_{\text{sum}} = 2.5$  bit/s/Hz and  $C_m = \frac{C_{\text{sum}}}{M}$  and  $|\mathbb{X}| = 4$ ,  $|\mathbb{Y}_m| = 64$ ,  $|\mathbb{Z}_m| = 4$

## 5.5 Discussion

In order to close the gap between the non-cooperative CEO scenario and the fully cooperative CEO scenario, this chapter introduced the pcCEO scenario, allowing partial cooperation among sensors in the network via rate-limited inter-sensor links. In particular, sensors are allowed to exchange some instantaneous side-information during run-time. Therefore, the GDIB algorithm has been extended for three different inter-sensor communication protocols: successive broadcasting, sequential point-to-point communication, and a two-phase transmission protocol. The successive broadcasting and the sequential point-to-point communication protocols do not separate the forwarding of instantaneous side-information and the forwarding of compressed information to the common receiver in distinct phases. Thus, both protocols can only exploit instantaneous side-information of previous sensors in the optimization chain. More precisely, the successive broadcasting protocol exploits the instantaneous side-information of all previous sensors. It turned out that in this way, the pcCEO scenario significantly outperforms the original CEO scenario, which does not allow cooperation among sensors. However, since the successive broadcasting protocol suffers from the curse of dimensionality for larger networks, the sequential point-to-point protocol has been introduced. Here, the instantaneous side-information is only forwarded to the next sensor in the optimization chain. By choosing an appropriate cardinality, the instantaneous side-information can contain the same amount of information about the relevant signal as in the broadcast case. Therefore, the overall performance is very similar to the performance of the successive broadcasting protocol. In addition, it can be observed that the pcCEO scenario shows a larger robustness to suboptimal Wyner-Ziv coding strategies in asymmetric scenarios. Naturally, this can be explained by sensors

being able to forward their good measurements to the neighboring sensor, although they might have a bad forward channel to the common receiver. However, for both transmission protocols, there still remains a performance gap to the fcCEO scenario. In order to close this gap, a third transmission protocol has been introduced, which separates the cooperation among sensors and the actual forwarding to the common receiver in distinct phases, as is the case in the fcCEO scenario. This allows each sensor to have access to the maximum available side-information. This communication protocol has been investigated pursuing the EXIT chart philosophy by interpreting the instantaneous side-information as extrinsic information available at each sensor. It turned out that the separation of cooperation and forwarding phase can close the gap to the fcCEO scenario. Although no formal conclusion about the optimality of the pcCEO scenario can be drawn, the closeness to the fcCEO scenario in the investigated simulations reveals that solutions found by the proposed greedy algorithms are at least close to optimal.



## Chapter 6

### Conclusion

This thesis considers a special case of the multi-terminal source coding problem, where all terminals, i.e., sensors, are interested in the same source signal while trying to forward their noisy measurements to a common receiver via rate-limited forward links. Therefore, each sensor has to locally compress its measurements. This scenario is widely known as the CEO problem [BZV96] and has been investigated for many different scenarios and model assumptions in the current literature. Most of the research is interested in finding the rate region from a theoretical point of view, which means trying to find the rates for a specific target distortion. This thesis, however, considers the problem in a different way, i.e., it tries to minimize the distortion for specific target rates. In particular, it provides an algorithmical perspective on this problem. Although evolving from a different field, the CEO problem with a logarithmic loss distortion measure is closely related to the information bottleneck principle. Therefore, this thesis applies the IB principle to design algorithmic approaches allowing the optimization of the compression device of each sensor, such that the distortion is minimized while not exceeding individual rate constraints.

Chapter 2 and 3 provide necessary theoretical background information to understand the main contributions of this thesis. Therefore, information-theoretic fundamentals, as well as basics on general source coding with lossy compression, have been discussed. The main focus here lies on the introduction of the information bottleneck principle as well as some properties and algorithmic approaches to solve it. Afterward, the non-cooperative distributed remote source coding problem, also known as the CEO problem, with a logarithmic loss distortion measure is introduced. In addition, the fully cooperative CEO scenario is described, assuming that each sensor has access to all measurements in the network.

The main contribution of this thesis can be found in Chapter 4 and 5. The GDIB algorithm has been introduced in the first part of Chapter 4 as an iterative algorithm, which sequentially designs the quantizer of each sensor applying the Wyner-Ziv coding principle by exploiting the statistics of previously designed quantizers [SK21; Ste+21b]. In this way, this algorithm maximizes the relevant mutual information while not exceeding the individual forward link capacities of each sensor. In contrast to approaches based on the sum-rate constraint, this allows the design even for asymmetric scenarios, where sensors have different link capacities and measurement SNRs. It has been shown that the GDIB algorithm outperforms individual scalar IB optimization for each sensor. In asymmetric scenarios, the GDIB algorithm is very sensitive to different Wyner-Ziv coding strategies, i.e., different optimization orders. However, a good solution can be expected by starting the optimization with the sensor belonging to the best forward link. It has been shown that the GDIB algorithm is quite robust against measurement SNR mismatch for good forward

links. In bad forward links, measurement SNR mismatch can cause the rate constraints to be violated. In order to prevent that, a compression rate back-off can be included, or the optimization can be performed for a low SNR. By setting the trade-off parameter to zero, a slightly modified variant of the GDIB algorithm results in deterministic mappings, which might be interesting for a practical implementation. However, since the compression can only be adjusted in coarse steps by changing the output cardinality, these deterministic mappings may cause performance losses. Since the memory complexity of the GDIB algorithm depends on the number of sensors in the network, it might be infeasible for larger networks. The second part of Chapter 4 introduced a reduced-memory complexity version of the GDIB algorithm allowing the optimization even for larger network sizes [Ste+21a; Ste+21b]. Therefore, the IB principle is applied in a sequential compression scheme to compress the mappings of previously designed quantizers to a single index while preserving information about the relevant signal. It has been shown that the performance loss due to the additional IB compression can be negligible by choosing appropriate cardinalities. Moreover, this sequential compression scheme significantly reduces the memory complexity. In the last part of Chapter 4, the GDIB algorithm has been adapted to incorporate imperfect erroneous forward links based on the approach in [Win14; WMB13]. This CA-GDIB algorithm has been shown to outperform the original GDIB algorithm for a scenario with a residual error probability on the forward links.

Chapter 4 demonstrated that although applying Wyner-Ziv coding with the GDIB algorithm, there remains a large gap between the non-cooperative CEO scenario and the fcCEO scenario where each sensor has access to all measurements. Therefore, Chapter 5 introduces the pcCEO scenario, allowing partial cooperation among sensors in the network during run-time via rate-limited inter-sensor links [SAK22; SK22]. The GDIB algorithm has been extended for three different inter-sensor communication protocols: successive broadcasting, sequential point-to-point communication, and a two-phase transmission protocol. In the successive broadcasting protocol, sensors are allowed to forward instantaneous side-information to all subsequent sensors. Since cooperation among sensors and forwarding to the common receiver is not performed in distinct phases, each sensor can only exploit the instantaneous side-information of all previous sensors in the optimization chain. It has been shown that this can significantly increase the performance compared to the non-cooperative CEO scenario. However, since this transmission protocol suffers from the curse of dimensionality, the sequential point-to-point protocol has been introduced as an alternative approach. Here, the instantaneous side-information can only be forwarded to the next sensor within the optimization chain. It has been shown that the performance of this transmission protocol nearly equals the performance of the successive broadcasting protocol if the output cardinality of the instantaneous side-information is chosen appropriately. Moreover, the exchange of instantaneous side-information during run-time increases the robustness against bad Wyner-Ziv coding strategies since good measurements can be forwarded to the neighboring sensor even if the forward channel to the common receiver is bad. However, there still occurs a performance gap to the fcCEO scenario. Therefore, the two-phase transmission protocol has been introduced, separating the cooperation among sensors and the transmission to the common receiver in distinct phases. In this case, each

sensor has access to the maximum available side-information. It has been shown that, in this case, the performance of the fcCEO scenario can be achieved.

## Bibliography

- [Ale+16] Alexander A. Alemi et al. “Deep Variational Information Bottleneck.” In: (2016). DOI: 10.48550/ARXIV.1612.00410. URL: <https://arxiv.org/abs/1612.00410>.
- [Ari72] S. Arimoto. “An algorithm for computing the capacity of arbitrary discrete memoryless channels.” In: *IEEE Transactions on Information Theory* 18.1 (1972), pp. 14–20.
- [AZ21] I. E. Aguerri and A. Zaidi. “Distributed Variational Representation Learning.” In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43.1 (2021), pp. 120–138. DOI: 10.1109/TPAMI.2019.2928806.
- [AZG02] A. Aaron, Rui Zhang, and B. Girod. “Wyner-Ziv coding of motion video.” In: *Conference Record of the Thirty-Sixth Asilomar Conference on Signals, Systems and Computers, 2002*. Vol. 1. 2002, 240–244 vol.1. DOI: 10.1109/ACSSC.2002.1197184.
- [Bau+18] G. Bauch et al. “Information-Optimum Discrete Signal Processing for Detection and Decoding - Invited Paper.” In: *2018 IEEE 87th Vehicular Technology Conference (VTC Spring)*. June 2018, pp. 1–6. DOI: 10.1109/VTCSpring.2018.8417508.
- [BBD22] Edgar Beck, Carsten Bockelmann, and Armin Dekorsy. *Semantic Communication: An Information Bottleneck View*. 2022. DOI: 10.48550/ARXIV.2204.13366. URL: <https://arxiv.org/abs/2204.13366>.
- [Ber71] Toby Berger. *Rate Distortion Theory: A Mathematical Basis for Data Compression*. Prentice-Hall, 1971.
- [Ber75] Toby Berger. “Rate Distortion Theory and Data Compression.” In: *Advances in Source Coding*. Vienna: Springer Vienna, 1975, pp. 1–39. ISBN: 978-3-7091-2928-9. DOI: 10.1007/978-3-7091-2928-9\_1. URL: [https://doi.org/10.1007/978-3-7091-2928-9\\_1](https://doi.org/10.1007/978-3-7091-2928-9_1).
- [BG98] T. Berger and J. D. Gibson. “Lossy source coding.” In: *IEEE Transactions on Information Theory* 44.6 (1998), pp. 2693–2723. DOI: 10.1109/18.720552.
- [Bis06] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN: 0387310738.
- [Bla72] R. Blahut. “Computation of channel capacity and rate-distortion functions.” In: *IEEE Transactions on Information Theory* 18.4 (1972), pp. 460–473.
- [Bri01] S. ten Brink. “Convergence Behavior of Iteratively Decoded Parallel Concatenated Codes.” In: *IEEE Transactions on Communications* 49.10 (Oct. 2001), pp. 1727–1737.

- 
- [BV04] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004. ISBN: 9781107394001. URL: <https://books.google.de/books?id=IUZdAAAAQBAJ>.
- [BZV96] T. Berger, Z. Zhang, and H. Viswanathan. “The CEO Problem [Multiterminal Source Coding].” In: *IEEE Transactions on Information Theory* 42.3 (May 1996), pp. 887–902. ISSN: 0018-9448. DOI: 10.1109/18.490552. URL: <https://doi.org/10.1109/18.490552>.
- [Che+03] Gal Chechik et al. “Information Bottleneck for Gaussian Variables.” In: *Advances in Neural Information Processing Systems*. Ed. by S. Thrun, L. Saul, and B. Schölkopf. Vol. 16. MIT Press, 2003. URL: <https://proceedings.neurips.cc/paper/2003/file/7e05d6f828574fbc975a896b25bb011e-Paper.pdf>.
- [Che+15] Aleksandra Checko et al. “Cloud RAN for Mobile Networks—A Technology Overview.” In: *IEEE Communications Surveys & Tutorials* 17.1 (2015), pp. 405–426. DOI: 10.1109/COMST.2014.2355255.
- [CK16a] D. Chen and V. Kuehn. “Alternating information bottleneck optimization for the compression in the uplink of C-RAN.” In: *2016 IEEE International Conference on Communications (ICC)*. May 2016, pp. 1–7. DOI: 10.1109/ICC.2016.7510694.
- [CK16b] D. Chen and V. Kuehn. “Alternating Information Bottleneck Optimization for Weighted Sum Rate and Resource Allocation in the Uplink of C-RAN.” In: *WSA 2016; 20th International ITG Workshop on Smart Antennas*. Mar. 2016, pp. 1–7.
- [CT06] T.M. Cover and J.A. Thomas. *Elements of Information Theory*. second. New York: Wiley & Sons, 2006.
- [CT84] Imre Csiszar and Gábor Tusnády. “Information geometry and alternating minimization procedures.” In: *Stat Decis* 1 (Jan. 1984).
- [CW11] J. Chen and J. Wang. “On the vector Gaussian CEO problem.” In: *2011 IEEE International Symposium on Information Theory Proceedings*. July 2011, pp. 2050–2054. DOI: 10.1109/ISIT.2011.6033916.
- [CW14] T. A. Courtade and T. Weissman. “Multiterminal Source Coding Under Logarithmic Loss.” In: *IEEE Transactions on Information Theory* 60.1 (2014), pp. 740–761.
- [DT62] R. Dobrushin and B. Tsybakov. “Information transmission with additional noise.” In: *IRE Transactions on Information Theory* 8.5 (1962), pp. 293–304. DOI: 10.1109/TIT.1962.1057738.
- [DW04] S.C. Draper and G.W. Wornell. “Side Information Aware Coding Strategies for Sensor Networks.” In: *IEEE Journal on Selected Areas in Communications* 22.6 (Aug. 2004), pp. 966–976.
-

- 
- [EG18] K. Eswaran and M. Gastpar. “Remote Source Coding under Gaussian Noise: Dueling Roles of Power and Entropy Power.” In: *arxiv preprint* (2018). URL: <https://arxiv.org/abs/1805.06515v2>.
- [EG88] Y. Ephraim and R.M. Gray. “A unified approach for encoding clean and noisy sources by means of waveform and autoregressive model vector quantization.” In: *IEEE Transactions on Information Theory* 34.4 (1988), pp. 826–834. DOI: 10.1109/18.9780.
- [EK11] A. El Gamal and Y.-H. Kim. *Network Information Theory*. Cambridge University Press, 2011.
- [EZ18] Iñaki Estella Aguerri and Abdellatif Zaidi. “Distributed information bottleneck method for discrete and Gaussian sources.” In: *The International Zurich Seminar on Information and Communication (IZS 2018) Proceedings*. ETH Zurich. 2018, pp. 35–39.
- [FD07] Brendan J. Frey and Delbert Dueck. “Clustering by Passing Messages Between Data Points.” In: *Science* 315.5814 (2007), pp. 972–976. DOI: 10.1126/science.1136800. eprint: <https://www.science.org/doi/pdf/10.1126/science.1136800>. URL: <https://www.science.org/doi/abs/10.1126/science.1136800>.
- [FK19] D. Franz and V. Kühn. “Vector Approximate Message Passing using Information Bottleneck Optimized Lookup Tables.” In: *12th International ITG Conference on Systems, Communications and Coding*. Rostock, Germany, Feb. 2019.
- [Fuj05] Satoru Fujishige. *Submodular functions and optimization*. Elsevier, 2005.
- [GNT03] Ran Gilad-Bachrach, Amir Navot, and Naftali Tishby. “An Information Theoretic Tradeoff between Complexity and Accuracy.” In: *Lecture Notes in Computer Science* (July 2003). DOI: 10.1007/978-3-540-45167-9\_43.
- [GPD12] T. Gedeon, A. E Parker, and A. G Dimitrov. “The mathematical structure of information bottleneck methods.” In: *Entropy* 14.3 (2012), pp. 456–479.
- [Gra90] Robert M. Gray. “Distortion-Rate Theory.” In: *Source Coding Theory*. Boston, MA: Springer US, 1990, pp. 53–79. ISBN: 978-1-4613-1643-5. DOI: 10.1007/978-1-4613-1643-5\_3. URL: [https://doi.org/10.1007/978-1-4613-1643-5\\_3](https://doi.org/10.1007/978-1-4613-1643-5_3).
- [GS09] C.M. Grinstead and J.L. Snell. *Grinstead and Snell’s Introduction to Probability*. Titolo collana. University Press of Florida, 2009. ISBN: 9781616100469. URL: <https://books.google.de/books?id=cE6FRAAACA AJ>.
- [GVD06] M. Gastpar, M. Vetterli, and P.L. Dragotti. “Sensing Reality and Communicating Bits: A Dangerous Liaison.” In: *IEEE Signal Processing Magazine* (July 2006), pp. 70–83.

- 
- [Has+17] S. Hassanpour et al. “On the relation between the asymptotic performance of different algorithms for information bottleneck framework.” In: *2017 IEEE International Conference on Communications (ICC)*. May 2017, pp. 1–6. DOI: 10.1109/ICC.2017.7996419.
- [Has+20] Shayan Hassanpour et al. “Forward-Aware Information Bottleneck-Based Vector Quantization for Noisy Channels.” In: *IEEE Transactions on Communications* 68.12 (2020), pp. 7911–7926. DOI: 10.1109/TCOMM.2020.3019447.
- [HK80] T.S. Han and K. Kobayashi. “A Unified Achievable Rate Region for a General Class of Multiterminal Source Coding Systems.” In: *IEEE Transactions on Information Theory* IT-26.3 (May 1980), pp. 277–288.
- [HL04] J.B. Hiriart-Urruty and C. Lemaréchal. *Fundamentals of Convex Analysis*. Grundlehren Text Editions. Springer Berlin Heidelberg, 2004. ISBN: 9783540422051. URL: [https://books.google.de/books?id=Ben6nm%5C\\_yapMC](https://books.google.de/books?id=Ben6nm%5C_yapMC).
- [HT07] Peter Harremoës and Naftali Tishby. “The Information Bottleneck Revisited or How to Choose a Good Distortion Measure.” In: *2007 IEEE International Symposium on Information Theory*. 2007, pp. 566–570. DOI: 10.1109/ISIT.2007.4557285.
- [Huf52] David A. Huffman. “A Method for the Construction of Minimum-Redundancy Codes.” In: *Proceedings of the IRE* 40.9 (1952), pp. 1098–1101. DOI: 10.1109/JRPROC.1952.273898.
- [HWD17] Shayan Hassanpour, Dirk Wuebben, and Armin Dekorsy. “Overview and Investigation of Algorithms for the Information Bottleneck Method.” In: *proceedings: SCC*. 2017.
- [HWD18a] S. Hassanpour, D. Wübben, and A. Dekorsy. “A Graph-Based Message Passing Approach for Joint Source-Channel Coding via Information Bottleneck Principle.” In: *10th Int. Symposium on Turbo Codes & Iterative Information Processing (ISTC 2018)*. Hong Kong, China, Dec. 2018. URL: <http://www.istc2018.org/>.
- [HWD18b] S. Hassanpour, D. Wübben, and A. Dekorsy. “A Graph-Based Message Passing Approach for Noisy Source Coding via Information Bottleneck Principle.” In: *IEEE Global Communications Conference (GLOBECOM 2018)*. Abu Dhabi, United Arab Emirates, Dec. 2018. URL: <http://globecom2018.ieee-globecom.org/>.
- [HWD18c] S. Hassanpour, D. Wübben, and A. Dekorsy. “On the Equivalence of Double Maxima and KL-Means for Information Bottleneck-Based Source Coding.” In: *IEEE Wireless Communications and Networking Conference (WCNC 2018)*. Barcelona, Spain, Apr. 2018. URL: <http://wcnc2018.ieee-wcnc.org/>.
-

- 
- [HWD18d] S. Hassanpour, D. Wübben, and A. Dekorsy. “On the Equivalence of Two Information Bottleneck-Based Routines Devised for Joint Source-Channel Coding.” In: *25th Int. Conference on Telecommunication (ICT 2018)*. Saint-Malo, France, May 2018. URL: <http://ict-2018.org/>.
- [HWD19] S. Hassanpour, D. Wübben, and A. Dekorsy. “A Novel Approach to Distributed Quantization via Multivariate Information Bottleneck Method.” In: *IEEE Global Communications Conference (GLOBECOM 2019)*. Waikoloa, HI, USA, Dec. 2019. URL: <https://globecom2019.ieee-globecom.org/>.
- [HWD20] S. Hassanpour, D. Wübben, and A. Dekorsy. “Generalized Distributed Information Bottleneck for Fronthaul Rate Reduction at the Cloud-RANs Uplink.” In: *IEEE Global Communications Conference (GLOBECOM 2020)*. Taipei, Taiwan, Dec. 2020. URL: <https://globecom2020.ieee-globecom.org/>.
- [HWD21] Shayan Hassanpour, Dirk Wübben, and Armin Dekorsy. “Forward-Aware Information Bottleneck-Based Vector Quantization: Multiterminal Extensions for Parallel and Successive Retrieval.” In: *IEEE Transactions on Communications* 69.10 (2021), pp. 6633–6646. DOI: 10.1109/TCOMM.2021.3097142.
- [Jun+04] Jun Chen et al. “An upper bound on the sum-rate distortion function and its corresponding rate allocation schemes for the CEO problem.” In: *IEEE Journal on Selected Areas in Communications* 22.6 (Aug. 2004), pp. 977–987. ISSN: 1558-0008. DOI: 10.1109/JSAC.2004.830888.
- [Kie93] J.C. Kieffer. “A survey of the theory of source coding with a fidelity criterion.” In: *IEEE Transactions on Information Theory* 39.5 (1993), pp. 1473–1490. DOI: 10.1109/18.259634.
- [KK17a] D. Kern and V. Kuehn. “On Implicit and Explicit Channel Estimation for Compress and Forward Relaying OFDM Schemes Designed by Information Bottleneck Graphs.” In: *WSA 2017; 21th International ITG Workshop on Smart Antennas*. Mar. 2017, pp. 1–8.
- [KK17b] D. Kern and V. Kühn. “On Compress and Forward with Multiple Carriers in the 3-Node Relay Channel Exploiting Information Bottleneck Graphs.” In: *11th International ITG Conference on Systems, Communications and Coding (SCC 2017)*. Hamburg, Germany, Feb. 2017.
- [KK17c] D. Kern and V. Kühn. “On Information Bottleneck Graphs to Design Compress and Forward Quantizers Exploiting Side Information for Multi-Carrier Transmission.” In: *IEEE International Conference on Communications*. Paris, France, May 2017.
- [Kol33] A.N. Kolmogorov. *Grundbegriffe der Wahrscheinlichkeitsrechnung*. Ergebnisse der Mathematik und ihrer Grenzgebiete. J. Springer, 1933. URL: <https://books.google.de/books?id=ob4rAAAAYAAJ>.
-



- 
- [Kol56] A. Kolmogorov. “On the Shannon theory of information transmission in the case of continuous signals.” In: *IRE Transactions on Information Theory* 2.4 (1956), pp. 102–108. DOI: 10.1109/TIT.1956.1056823.
- [Kur17] B. M. Kurkoski. “On the Relationship Between the KL Means Algorithm and the Information Bottleneck Method.” In: *SCC 2017; 11th International ITG Conference on Systems, Communications and Coding*. 2017, pp. 1–6.
- [KY14] Brian M. Kurkoski and Hideki Yagi. “Quantization of Binary-Input Discrete Memoryless Channels.” In: *IEEE Transactions on Information Theory* 60.8 (2014), pp. 4544–4552. DOI: 10.1109/TIT.2014.2327016.
- [LB15] Jan Lewandowsky and G. Bauch. “Trellis Based Node Operations for LDPC Decoders from the Information Bottleneck Method.” In: *2015 9th International Conference on Signal Processing and Communication Systems (ICSPCS)*. IEEE, Jan. 2015, pp. 1–10. ISBN: 978-1-4673-8118-5. DOI: 10.1109/ICSPCS.2015.7391731.
- [LB18] J. Lewandowsky and G. Bauch. “Information-Optimum LDPC Decoders Based on the Information Bottleneck Method.” In: *IEEE Access* 6 (2018), pp. 4054–4071. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2018.2797694.
- [Lew+17] J. Lewandowsky et al. “Discrete Channel Estimation by Integer Passing in Information Bottleneck Graphs.” In: *SCC 2017; 11th International ITG Conference on Systems, Communications and Coding*. Feb. 2017, pp. 1–6.
- [Lew+18] Jan Lewandowsky et al. “Design and Evaluation of Information Bottleneck LDPC Decoders for Software Defined Radios.” In: *2018 12th International Conference on Signal Processing and Communication Systems (ICSPCS)*. 2018, pp. 1–9. DOI: 10.1109/ICSPCS.2018.8631719.
- [Lin91] J. Lin. “Divergence measures based on the Shannon entropy.” In: *IEEE Transactions on Information Theory* 37.1 (1991), pp. 145–151. DOI: 10.1109/18.61115.
- [LSB16a] Jan Lewandowsky, Maximilian Stark, and Gerhard Bauch. “Information Bottleneck Graphs for Receiver Design.” In: *Proc. IEEE ISIT’16*. IEEE, Jan. 1, 2016, pp. 2888–2892. ISBN: 978-1-5090-1806-2. DOI: 10.1109/ISIT.2016.7541827.
- [LSB16b] Jan Lewandowsky, Maximilian Stark, and Gerhard Bauch. “Optimum message mapping LDPC decoders derived from the sum-product algorithm.” In: *2016 IEEE International Conference on Communications (ICC)*. 2016, pp. 1–6. DOI: 10.1109/ICC.2016.7510906.
- [LSB17] J. Lewandowsky, M. Stark, and G. Bauch. “Message alignment for discrete LDPC decoders with quadrature amplitude modulation.” In: *IEEE ISIT’17*. IEEE ISIT’17. June 2017, pp. 2925–2929. DOI: 10.1109/ISIT.2017.8007065.
-

- 
- [MA16] Sahar Movaghati and Masoud Ardakani. “Distributed Channel-Aware Quantization Based on Maximum Mutual Information.” In: *International Journal of Distributed Sensor Networks* 12.5 (2016), p. 3595389. DOI: 10.1155/2016/3595389.
- [McC05] S. Thomas McCormick. “Submodular Function Minimization.” In: *Discrete Optimization*. Ed. by K. Aardal, G.L. Nemhauser, and R. Weismantel. Vol. 12. Handbooks in Operations Research and Management Science. Elsevier, 2005, pp. 321–391. DOI: [https://doi.org/10.1016/S0927-0507\(05\)12007-6](https://doi.org/10.1016/S0927-0507(05)12007-6). URL: <https://www.sciencedirect.com/science/article/pii/S0927050705120076>.
- [MSK22] Clemens- Konrad Mueller, Steffen Steiner, and Volker Kuehn. “Information Bottleneck Optimization of a Finite Bit Resolution Fast Fourier Transform.” In: *2022 IEEE Symposium on Computers and Communications (ISCC)*. 2022, pp. 1–6. DOI: 10.1109/ISCC55528.2022.9913001.
- [MWD19] Tobias Monsees, Dirk Wuebben, and Armin Dekorsy. “Channel-Optimized Information Bottleneck Design for Signal Forwarding and Discrete Decoding in Cloud-RAN.” In: *SCC 2019; 12th International ITG Conference on Systems, Communications and Coding*. 2019, pp. 1–6. DOI: 10.30420/454862041.
- [Ooh05] Y. Oohama. “Rate-distortion theory for Gaussian multiterminal source coding systems with several side informations at the decoder.” In: *IEEE Transactions on Information Theory* 51.7 (2005), pp. 2577–2593. DOI: 10.1109/TIT.2005.850110.
- [Ooh08] Y. Oohama. “Distributed Source Coding of Correlated Gaussian Observations.” In: *International Symposium on Information Theory and its Applications (ISITA 2008)*. Auckland, New Zealand, Dec. 2008, pp. 119–.
- [Ooh12] Y. Oohama. “Distributed Source Coding of Correlated Gaussian Remote Sources.” In: *IEEE Transactions on Information Theory* 58.8 (Aug. 2012).
- [Ooh98] Y. Oohama. “The rate-distortion function for the quadratic Gaussian CEO problem.” In: *IEEE Transactions on Information Theory* 44.3 (May 1998), pp. 1057–1070. ISSN: 1557-9654. DOI: 10.1109/18.669162.
- [Par+14] S.-H. Park et al. “Fronthaul Compression for Cloud Radio Access Networks: Signal Processing Advances Inspired by Network Information Theory.” In: *IEEE Signal Processing Magazine* 31.6 (Nov. 2014), pp. 69–79.
- [PP02] A. Papoulis and S.U. Pillai. *Probability, Random Variables, and Stochastic Processes*. McGraw-Hill series in electrical and computer engineering. McGraw-Hill, 2002. ISBN: 9780071226615. URL: <https://books.google.de/books?id=cUmiDAEACAAJ>.
- [PRT04] V. Prabhakaran, K. Ramchandran, and D. Tse. “On the Role of Interaction Between Sensors in the CEO Problem.” In: *42nd Annual Allerton Conference on Communication, Control, and Computing*. 2004.
-

- 
- [PSW09] H. Permuter, Y. Steinberg, and T. Weissman. “Problems we can solve with a helper.” In: *2009 IEEE Information Theory Workshop on Networking and Information Theory*. 2009, pp. 266–270. DOI: 10.1109/ITWNET.2009.5158584.
- [PTR04] V. Prabhakaran, D. Tse, and K. Ramachandran. “Rate region of the quadratic Gaussian CEO problem.” In: *International Symposium on Information Theory, 2004. ISIT 2004. Proceedings*. June 2004, pp. 119–. DOI: 10.1109/ISIT.2004.1365154.
- [SAK22] Steffen Steiner, Abdulrahman Dayo Aminu, and Volker Kuehn. “Distributed Quantization for Partially Cooperating Sensors Using the Information Bottleneck Method.” In: *Entropy* 24.4 (2022). ISSN: 1099-4300. DOI: 10.3390/e24040438. URL: <https://www.mdpi.com/1099-4300/24/4/438>.
- [Sak68] D. Sakrison. “Source encoding in the presence of random disturbance.” In: *IEEE Transactions on Information Theory* 14.1 (Jan. 1968), pp. 165–167.
- [Sax+19] Andrew M Saxe et al. “On the information bottleneck theory of deep learning.” In: *Journal of Statistical Mechanics: Theory and Experiment* 2019.12 (2019), p. 124020. DOI: 10.1088/1742-5468/ab3985. URL: <https://dx.doi.org/10.1088/1742-5468/ab3985>.
- [Sch+01] Elad Schneidman et al. “Analyzing Neural Codes Using the Information Bottleneck Method.” In: *Submitted to NIPS* (July 2001). URL: [https://www.cs.huji.ac.il/labs/learning/Papers/nips01\\_sub.pdf](https://www.cs.huji.ac.il/labs/learning/Papers/nips01_sub.pdf).
- [SFT06] Noam Slonim, Nir Friedman, and Naftali Tishby. “Multivariate Information Bottleneck.” In: *Neural Computation* 18.8 (2006), pp. 1739–1789. DOI: 10.1162/neco.2006.18.8.1739.
- [Sha48] C.E. Shannon. “A mathematical theory of communication.” In: *The Bell System Technical Journal* 27.3 (July 1948), pp. 379–423.
- [Sha59] C.E. Shannon. “Coding theorems for a discrete source with a fidelity criterion.” In: *Institute of Radio Engineers, International Convention Record* 7 (1959), pp. 142–163.
- [Sim09] O. Simeone. “Source and Channel Coding for Homogeneous Sensor Networks with Partial Cooperation.” In: *IEEE Transactions on Wireless Communications* 8.3 (Mar. 2009).
- [SK19] Steffen Steiner and Volker Kuehn. “Optimization Of Distributed Quantizers Using An Alternating Information Bottleneck Approach.” In: *WSA 2019; 23rd International ITG Workshop on Smart Antennas*. 2019, pp. 1–6.
- [SK21] Steffen Steiner and Volker Kuehn. “Distributed Compression using the Information Bottleneck Principle.” In: *ICC 2021 - IEEE International Conference on Communications*. 2021, pp. 1–6. DOI: 10.1109/ICC42927.2021.9500324.

- 
- [SK22] Steffen Steiner and Volker Kuehn. “Distributed Compression for Partially Cooperating Sensors and Gaussian Relevant Signals.” In: *International ITG 26th Workshop on Smart Antennas and 13th Conference on Systems, Communications, and Coding*. 2022, pp. 1–6.
- [SLB18a] Maximilian Stark, Jan Lewandowsky, and Gerhard Bauch. “Information-Bottleneck Decoding of High-Rate Irregular LDPC Codes for Optical Communication Using Message Alignment.” In: *Applied Sciences* 8.10 (2018). ISSN: 2076-3417. DOI: 10.3390/app8101884. URL: <https://www.mdpi.com/2076-3417/8/10/1884>.
- [SLB18b] Maximilian Stark, Jan Lewandowsky, and Gerhard Bauch. “Information-Optimum LDPC Decoders with Message Alignment for Irregular Codes.” In: *2018 IEEE Global Communications Conference (GLOBECOM)*. 2018, pp. 1–6. DOI: 10.1109/GLOCOM.2018.8648053.
- [SLB19] Maximilian Stark, Jan Lewandowsky, and Gerhard Bauch. “A Parametric Information Bottleneck Algorithm for Gaussian Random Variables and Gaussian Mixtures.” In: *SCC 2019; 12th International ITG Conference on Systems, Communications and Coding*. 2019, pp. 1–6. DOI: 10.30420/454862009.
- [SLB20] Maximilian Stark, Jan Lewandowsky, and Gerhard Bauch. “Neural Information Bottleneck Decoding.” In: *2020 14th International Conference on Signal Processing and Communication Systems (ICSPCS)*. 2020, pp. 1–7. DOI: 10.1109/ICSPCS50536.2020.9310014.
- [Slo02] N. Slonim. “The Information Bottleneck Theory and Applications.” PhD thesis. Hebrew University of Jerusalem, Jan. 2002.
- [SS17] DJ Strouse and David J. Schwab. “The Deterministic Information Bottleneck.” In: *Neural Computation* 29.6 (2017), pp. 1611–1630. DOI: 10.1162/NECO\_a\_00961.
- [SSB18] M. Stark, A. Shah, and G. Bauch. “Polar code construction using the information bottleneck method.” In: *2018 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*. Apr. 2018, pp. 7–12. DOI: 10.1109/WCNCW.2018.8368978.
- [SSB19] Syed Aizaz Ali Shah, Maximilian Stark, and Gerhard Bauch. “Coarsely Quantized Decoding and Construction of Polar Codes Using the Information Bottleneck Method.” In: *Algorithms* 12.9 (2019). ISSN: 1999-4893. DOI: 10.3390/a12090192. URL: <https://www.mdpi.com/1999-4893/12/9/192>.
- [ST00] Noam Slonim and Naftali Tishby. “Document Clustering Using Word Clusters via the Information Bottleneck Method.” In: *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR ’00. Athens, Greece: Association for Computing Machinery, 2000, pp. 208–215. ISBN: 1581132263. DOI: 10.1145/345508.345578. URL: <https://doi.org/10.1145/345508.345578>.
-

- 
- [ST99] Noam Slonim and Naftali Tishby. “Agglomerative Information Bottleneck.” In: *Advances in Neural Information Processing Systems*. Ed. by S. Solla, T. Leen, and K. Müller. Vol. 12. MIT Press, 1999. URL: <https://proceedings.neurips.cc/paper/1999/file/be3e9d3f7d70537357c67bb3f4086846-Paper.pdf>.
- [Sta+19] Maximilian Stark et al. “Decoding of Non-Binary LDPC Codes using the Information Bottleneck Method.” In: *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*. 2019, pp. 1–6. DOI: 10.1109/ICC.2019.8761712.
- [Sta+20a] Maximilian Stark et al. “Decoding Rate-Compatible 5G-LDPC Codes With Coarse Quantization Using the Information Bottleneck Method.” In: *IEEE Open Journal of the Communications Society* 1 (2020), pp. 646–660. DOI: 10.1109/OJCOMS.2020.2994048.
- [Sta+20b] Maximilian Stark et al. “Information Bottleneck Decoding of Rate-Compatible 5G-LDPC Codes.” In: *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*. 2020, pp. 1–6. DOI: 10.1109/ICC40277.2020.9149304.
- [Ste+21a] Steffen Steiner et al. “Reduced-Complexity Greedy Distributed Information Bottleneck Algorithm.” In: *2021 IEEE Statistical Signal Processing Workshop (SSP)*. 2021, pp. 361–365. DOI: 10.1109/SSP49050.2021.9513805.
- [Ste+21b] Steffen Steiner et al. “Reduced-Complexity Optimization of Distributed Quantization Using the Information Bottleneck Principle.” In: *IEEE Open Journal of the Communications Society* 2 (2021), pp. 1267–1278. DOI: 10.1109/OJCOMS.2021.3083569.
- [TPB99] N. Tishby, F. C. Pereira, and W. Bialek. “The Information Bottleneck Method.” In: *37th Annual Allerton Conference on Communication, Control, and Computing*. Sept. 1999, pp. 368–377.
- [TZ15] Naftali Tishby and Noga Zaslavsky. “Deep learning and the information bottleneck principle.” In: *2015 IEEE Information Theory Workshop (ITW)* (2015), pp. 1–5.
- [UAZ17] Yiğit Uğur, Inaki Estella Aguerri, and Abdellatif Zaidi. “A generalization of blahut-arimoto algorithm to compute rate-distortion regions of multiterminal source coding under logarithmic loss.” In: *2017 IEEE Information Theory Workshop (ITW)*. IEEE. 2017, pp. 349–353.
- [UAZ18] Yigit Ugur, Inaki Estella Aguerri, and Abdellatif Zaidi. “Vector Gaussian CEO problem under logarithmic loss.” In: *2018 IEEE Information Theory Workshop (ITW)*. IEEE. 2018, pp. 1–5.
-

- 
- [UAZ20a] Yigit Ugur, Inaki Aguerri, and Abdellatif Zaidi. “Vector Gaussian CEO Problem Under Logarithmic Loss and Applications.” In: *IEEE Transactions on Information Theory* PP (Feb. 2020), pp. 1–1. DOI: 10.1109/TIT.2020.2972348.
- [UAZ20b] Yiğit Uğur, George Arvanitakis, and Abdellatif Zaidi. “Variational Information Bottleneck for Unsupervised Clustering: Deep Gaussian Mixture Embedding.” In: *Entropy* 22.2 (2020). ISSN: 1099-4300. DOI: 10.3390/e22020213. URL: <https://www.mdpi.com/1099-4300/22/2/213>.
- [VB97] H. Viswanathan and T. Berger. “The Quadratic Gaussian CEO Problem.” In: *IEEE Transactions on Information Theory* 43.5 (Sept. 1997), pp. 1549–1559. ISSN: 0018-9448. DOI: 10.1109/18.623151. URL: <https://doi.org/10.1109/18.623151>.
- [VRP19] Matías Vera, Leonardo Rey Vega, and Pablo Piantanida. “Collaborative Information Bottleneck.” In: *IEEE Transactions on Information Theory* 65.2 (2019), pp. 787–815. DOI: 10.1109/TIT.2018.2883295.
- [VVP17] Matias Vera, Leonardo Rey Vega, and Pablo Piantanida. “Distributed cooperative information bottleneck.” In: *2017 IEEE International Symposium on Information Theory (ISIT)*. 2017, pp. 709–713. DOI: 10.1109/ISIT.2017.8006620.
- [WC12] J. Wang and J. Chen. “On the vector Gaussian L-terminal CEO problem.” In: *2012 IEEE International Symposium on Information Theory Proceedings*. July 2012, pp. 571–575. DOI: 10.1109/ISIT.2012.6284256.
- [Win14] A. Winkelbauer. “Blind Performance Estimation and Quantizer Design with Applications to Relay Networks.” PhD thesis. Technical University Vienna, Austria, Dec. 2014.
- [Wit80] H. Witsenhausen. “Indirect rate distortion problems.” In: *IEEE Transactions on Information Theory* 26.5 (1980), pp. 518–521. DOI: 10.1109/TIT.1980.1056251.
- [WMB13] Andreas Winkelbauer, Gerald Matz, and Andreas Burg. “Channel-optimized vector quantization with mutual information as fidelity criterion.” In: *2013 Asilomar Conference on Signals, Systems and Computers*. 2013, pp. 851–855. DOI: 10.1109/ACSSC.2013.6810408.
- [WTV08] A.B. Wagner, S. Tavildar, and P. Viswanath. “Rate Region of the Quadratic Gaussian Two-Encoder Source-Coding Problem.” In: *IEEE Transactions on Information Theory* 54.5 (May 2008), pp. 1938–1961.
- [WW75] H. Witsenhausen and A. Wyner. “A conditional entropy bound for a pair of discrete random variables.” In: *IEEE Transactions on Information Theory* 21.5 (1975), pp. 493–501. DOI: 10.1109/TIT.1975.1055437.
-

- 
- [Wyn78] Aaron D. Wyner. “The Rate-Distortion Function for Source Coding with Side Information at the Decoder-II. General Sources.” In: *Inf. Control.* 38 (1978), pp. 60–80.
- [WZ70] J. Wolf and J. Ziv. “Transmission of Noisy Information to a Noisy Receiver With Minimum Distortion.” In: *IEEE Transactions on Information Theory* IT-16.4 (July 1970), pp. 406–411.
- [WZ71] A. Wyner and J. Ziv. “Bounds on the rate-distortion function for stationary sources with memory.” In: *IEEE Transactions on Information Theory* 17.5 (1971), pp. 508–513. DOI: 10.1109/TIT.1971.1054699.
- [WZ76] A. Wyner and J. Ziv. “The rate-distortion function for source coding with side information at the decoder.” In: *IEEE Transactions on Information Theory* 22.1 (1976), pp. 1–10. DOI: 10.1109/TIT.1976.1055508.
- [Xio+10] Zhe-yuan Xiong et al. “Distributed image coding in wireless multimedia sensor networks: A survey.” In: *Third International Workshop on Advanced Computational Intelligence*. IEEE. 2010, pp. 618–622.
- [XW16] Y. Xu and Q. Wang. “Rate Region of the Vector Gaussian CEO Problem With the Trace Distortion Constraint.” In: *IEEE Transactions on Information Theory* 62.4 (Apr. 2016), pp. 1823–1835. ISSN: 1557-9654. DOI: 10.1109/TIT.2016.2531080.
- [ZA20] Abdellatif Zaidi and Inaki Estella Aguerri. “Distributed Deep Variational Information Bottleneck.” In: *2020 IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*. 2020, pp. 1–5. DOI: 10.1109/SPAWC48557.2020.9154315.
- [Zei10] G. Zeitler. “Low-precision analog-to-digital conversion and mutual information in channels with memory.” In: *Proceedings 48th Annual Allerton Conference on Communication, Control and Computing*. Sept. 2010, pp. 745–752. ISBN: 978-1-4244-8215-3. (Visited on 07/15/2014).
- [Zei12] Georg Zeitler. “Low-Precision Quantizer Design for Communication Problems.” Dissertation. Muenchen: Technische Universitaet Muenchen, 2012.
- [ZES20] Abdellatif Zaidi, Iñaki Estella-Aguerrri, and Shlomo Shamai (Shitz). “On the Information Bottleneck Problems: Models, Connections, Applications and Information Theoretic Views.” In: *Entropy* 22.2 (2020). ISSN: 1099-4300. DOI: 10.3390/e22020151. URL: <https://www.mdpi.com/1099-4300/22/2/151>.

## Own Publications

### Primary Author

- [SAK22] Steffen Steiner, Abdulrahman Dayo Aminu, and Volker Kuehn. “Distributed Quantization for Partially Cooperating Sensors Using the Information Bottleneck Method.” In: *Entropy* 24.4 (2022). ISSN: 1099-4300. DOI: 10.3390/e24040438. URL: <https://www.mdpi.com/1099-4300/24/4/438>.
- [SK19] Steffen Steiner and Volker Kuehn. “Optimization Of Distributed Quantizers Using An Alternating Information Bottleneck Approach.” In: *WSA 2019; 23rd International ITG Workshop on Smart Antennas*. 2019, pp. 1–6.
- [SK21] Steffen Steiner and Volker Kuehn. “Distributed Compression using the Information Bottleneck Principle.” In: *ICC 2021 - IEEE International Conference on Communications*. 2021, pp. 1–6. DOI: 10.1109/ICC42927.2021.9500324.
- [SK22] Steffen Steiner and Volker Kuehn. “Distributed Compression for Partially Cooperating Sensors and Gaussian Relevant Signals.” In: *International ITG 26th Workshop on Smart Antennas and 13th Conference on Systems, Communications, and Coding*. 2022, pp. 1–6.
- [Ste+21a] Steffen Steiner et al. “Reduced-Complexity Greedy Distributed Information Bottleneck Algorithm.” In: *2021 IEEE Statistical Signal Processing Workshop (SSP)*. 2021, pp. 361–365. DOI: 10.1109/SSP49050.2021.9513805.
- [Ste+21b] Steffen Steiner et al. “Reduced-Complexity Optimization of Distributed Quantization Using the Information Bottleneck Principle.” In: *IEEE Open Journal of the Communications Society* 2 (2021), pp. 1267–1278. DOI: 10.1109/OJCOMS.2021.3083569.

### Co-Author

- [MSK22] Clemens- Konrad Mueller, Steffen Steiner, and Volker Kuehn. “Information Bottleneck Optimization of a Finite Bit Resolution Fast Fourier Transform.” In: *2022 IEEE Symposium on Computers and Communications (ISCC)*. 2022, pp. 1–6. DOI: 10.1109/ISCC55528.2022.9913001.



# Appendices

## Appendix A

### Proof for Supermodularity of $I(\mathcal{Y}_S; \mathcal{Z}_S | \mathcal{Z}_{\bar{S}}, \mathcal{Q})$

On a finite set  $\mathbb{V}$  a function  $s : 2^{\mathbb{V}} \rightarrow \mathbb{R}$  is supermodular if for all  $A, B \subseteq \mathbb{V}$

$$s(A) + s(B) \leq s(A \cap B) + s(A \cup B) \quad (\text{A.1})$$

holds. Following the proof given in [CW14], the general definition for supermodular functions can be applied on the compression rates  $I(\mathcal{Y}_S; \mathcal{Z}_S | \mathcal{Z}_{\bar{S}}, \mathcal{Q})$  such that for  $A, B \subseteq S$

$$\begin{aligned} s(A) + s(B) &= I(\mathcal{Y}_A; \mathcal{Z}_A | \mathcal{Z}_{\bar{A}}, \mathcal{Q}) + I(\mathcal{Y}_B; \mathcal{Z}_B | \mathcal{Z}_{\bar{B}}, \mathcal{Q}) \\ &\stackrel{(a)}{=} H(\mathcal{Z}_A | \mathcal{Z}_{\bar{A}}, \mathcal{Q}) - H(\mathcal{Z}_A | \mathcal{Y}_A, \mathcal{Q}) + H(\mathcal{Z}_B | \mathcal{Z}_{\bar{B}}, \mathcal{Q}) - H(\mathcal{Z}_B | \mathcal{Y}_B, \mathcal{Q}) \\ &\stackrel{(b)}{=} H(\mathcal{Z}_A | \mathcal{Z}_{\bar{A}}, \mathcal{Q}) + H(\mathcal{Z}_B | \mathcal{Z}_{\bar{B}}, \mathcal{Q}) \\ &\quad - H(\mathcal{Z}_{A \cup B} | \mathcal{Y}_{A \cup B}, \mathcal{Q}) - H(\mathcal{Z}_{A \cap B} | \mathcal{Y}_{A \cap B}, \mathcal{Q}) \\ &\stackrel{(c)}{=} H(\mathcal{Z}_{A \setminus B} | \mathcal{Z}_{\bar{A}}, \mathcal{Q}) + H(\mathcal{Z}_{A \cap B} | \mathcal{Z}_{\bar{A} \cap \bar{B}}) \\ &\quad + H(\mathcal{Z}_B | \mathcal{Z}_{\bar{B}}, \mathcal{Q}) - H(\mathcal{Z}_{A \cup B} | \mathcal{Y}_{A \cup B}, \mathcal{Q}) - H(\mathcal{Z}_{A \cap B} | \mathcal{Y}_{A \cap B}, \mathcal{Q}) \\ &\stackrel{(d)}{=} H(\mathcal{Z}_{A \setminus B} | \mathcal{Z}_{\bar{A}}, \mathcal{Q}) + H(\mathcal{Z}_B | \mathcal{Z}_{\bar{B}}, \mathcal{Q}) - H(\mathcal{Z}_{A \cup B} | \mathcal{Y}_{A \cup B}, \mathcal{Q}) \\ &\quad + I(\mathcal{Y}_{A \cap B}; \mathcal{Z}_{A \cap B} | \mathcal{Z}_{\bar{A} \cap \bar{B}}, \mathcal{Q}) \\ &\stackrel{(e)}{\leq} H(\mathcal{Z}_{A \setminus B} | \mathcal{Z}_{\bar{A} \cup \bar{B}}, \mathcal{Q}) + H(\mathcal{Z}_B | \mathcal{Z}_{\bar{B}}, \mathcal{Q}) - H(\mathcal{Z}_{A \cup B} | \mathcal{Y}_{A \cup B}, \mathcal{Q}) \\ &\quad + I(\mathcal{Y}_{A \cap B}; \mathcal{Z}_{A \cap B} | \mathcal{Z}_{\bar{A} \cap \bar{B}}, \mathcal{Q}) \\ &\stackrel{(f)}{=} I(\mathcal{Y}_{A \cup B}; \mathcal{Z}_{A \cup B} | \mathcal{Z}_{\bar{A} \cup \bar{B}}, \mathcal{Q}) + I(\mathcal{Y}_{A \cap B}; \mathcal{Z}_{A \cap B} | \mathcal{Z}_{\bar{A} \cap \bar{B}}, \mathcal{Q}) \\ &= s(A \cup B) + s(A \cap B) \end{aligned} \quad (\text{A.2})$$

holds. Note that (a),(d), and (f) hold due to the Markov property given in (3.39). Moreover, different  $\mathcal{Z}_m$  are conditionally independent given  $\mathcal{Y}_m$  and  $\mathcal{Q}$ . Reordering different mappings  $p(z_m | y_m)$  results in (b). In (c), the chain rule of mutual information is applied. Finally, conditioning reduces the entropy leading to (e).

## Appendix B

### Derivation of the GDIB Algorithm

As stated in Subsection 4.1 the optimization problem of the GDIB algorithm for sensor  $m$  is given as

$$L_{\text{GDIB}} = \sum_{m=1}^M I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{<m}) - \beta_m \cdot I(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{<m})$$

By pursuing a greedy optimization approach, this formulation can be decomposed into  $M$  optimization problems:

$$\begin{aligned} L_{\text{GDIB}}^{(1)} &= I(\mathcal{X}; \mathcal{Z}_1) - \beta_1 I(\mathcal{Y}_1; \mathcal{Z}_1) \\ &\vdots \\ L_{\text{GDIB}}^{(M)} &= I(\mathcal{X}; \mathcal{Z}_M | \mathcal{Z}_{<M}) - \beta_M I(\mathcal{Y}_M; \mathcal{Z}_M | \mathcal{Z}_{<M}). \end{aligned}$$

The particular utility function for sensor  $m$  is given by

$$L_{\text{GDIB}}^{(m)}[p(z_m | y_m)] = I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{<m}) - \beta_m I(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{<m}), \quad (\text{B.2})$$

which can be solved by taking the derivative w.r.t. the mapping  $p(z_m | y_m)$  and equating it to zero. Note that the mapping  $p(z_m | y_m)$  obviously influences both terms,  $I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{<m})$  and  $I(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{<m})$ . In the following, the derivatives of both mutual information are given.

#### B.1 Derivative of $I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{<m})$

The relevant mutual information in (B.2) can be rewritten such that the desired mapping occurs explicitly.

$$\begin{aligned} I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{<m}) &= \mathbb{E}_{\mathcal{X}, \mathcal{Z}_{<m}} \left[ \log \frac{p(z_m | x, \mathbf{z}_{<m})}{p(z_m | \mathbf{z}_{<m})} \right] \\ &= \sum_{z_m} \sum_{y_m} p(z_m | y_m) \cdot \sum_x \sum_{\mathbf{z}_{<m}} p(x, y_m, \mathbf{z}_{<m}) \cdot \log \sum_{a \in \mathbb{Y}_m} p(z_m | a) p(a | \mathbf{z}_{<m}, x) \\ &\quad - \sum_{z_m} \sum_{y_m} p(z_m | y_m) \cdot \sum_{\mathbf{z}_{<m}} p(y_m, \mathbf{z}_{<m}) \cdot \log \sum_{a \in \mathbb{Y}_m} p(z_m | a) p(a | \mathbf{z}_{<m}) \quad (\text{B.3}) \end{aligned}$$

The derivative of (B.3) delivers

$$\begin{aligned}
 \frac{\partial I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{< m})}{\partial p(z_m | y_m)} &= \sum_x \sum_{\mathbf{z}_{< m}} p(x, y_m, \mathbf{z}_{< m}) \cdot \log p(z_m | x, \mathbf{z}_{< m}) \\
 &+ \sum_x \sum_{\mathbf{z}_{< m}} \underbrace{\left[ \sum_{y_m} p(z_m | y_m) \cdot p(x, y_m, \mathbf{z}_{< m}) \right]}_{=p(z_m, x, \mathbf{z}_{< m})} \cdot \frac{p(y_m | x, \mathbf{z}_{< m})}{p(z_m | x, \mathbf{z}_{< m})} \\
 &- \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{z}_{< m}) \cdot \log p(z_m | \mathbf{z}_{< m}) \\
 &- \sum_{\mathbf{z}_{< m}} \underbrace{\left[ \sum_{y_m} p(z_m | y_m) \cdot p(y_m, \mathbf{z}_{< m}) \right]}_{=p(z_m, \mathbf{z}_{< m})} \cdot \frac{p(y_m | \mathbf{z}_{< m})}{p(z_m | \mathbf{z}_{< m})} \\
 &= \sum_x \sum_{\mathbf{z}_{< m}} p(x, y_m, \mathbf{z}_{< m}) \log p(z_m | x, \mathbf{z}_{< m}) - \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{z}_{< m}) \log p(z_m | \mathbf{z}_{< m}) \\
 &= \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{z}_{< m}) \cdot \sum_x p(x | y_m, \mathbf{z}_{< m}) \cdot \log \frac{p(z_m | x, \mathbf{z}_{< m})}{p(z_m | \mathbf{z}_{< m})}. \tag{B.4}
 \end{aligned}$$

Exploiting Bayes' theorem, the argument of the logarithmic function can be rewritten to

$$\frac{p(z_m | x, \mathbf{z}_{< m})}{p(z_m | \mathbf{z}_{< m})} = \frac{p(x | \mathbf{z}_{\leq m})}{p(x | \mathbf{z}_{< m})} = \frac{p(x | \mathbf{z}_{\leq m})}{p(x | y_m, \mathbf{z}_{< m})} \cdot \frac{p(x | y_m, \mathbf{z}_{< m})}{p(x | \mathbf{z}_{< m})}. \tag{B.5}$$

The last ratio in (B.5) can be dropped because it does not depend on  $p(z_m | y_m)$  and its contribution can be incorporated into the Lagrange multiplier  $\beta_m$ . The insertion of the first ratio into (B.4) yields the contribution of the derivative of the relevant mutual information

$$\begin{aligned}
 \frac{\partial I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{< m})}{\partial p(z_m | y_m)} &\rightarrow - \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{z}_{< m}) \cdot \sum_x p(x | y_m, \mathbf{z}_{< m}) \cdot \log \frac{p(x | y_m, \mathbf{z}_{< m})}{p(x | \mathbf{z}_{\leq m})} \\
 &= - \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{z}_{< m}) \cdot D_{\text{KL}} [p(x | y_m, \mathbf{z}_{< m}) || p(x | \mathbf{z}_{\leq m})]. \tag{B.6}
 \end{aligned}$$

## B.2 Derivative of $I(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{< m})$

With the definition of the conditional compression rate

$$\begin{aligned}
 I(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{< m}) &= \mathbb{E}_{y_m, \mathbf{z}_{\leq m}} \left[ \log \frac{p(z_m | y_m)}{p(z_m | \mathbf{z}_{< m})} \right] \\
 &= \sum_{z_m} \sum_{y_m} p(z_m | y_m) p(y_m) \log p(z_m | y_m) \\
 &- \sum_{z_m} \sum_{y_m} p(z_m | y_m) \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{z}_{< m}) \log \sum_{a \in \mathbb{Y}_m} p(z_m | a) p(a | \mathbf{z}_{< m}), \tag{B.7}
 \end{aligned}$$

its derivative becomes

$$\begin{aligned}
 \frac{\partial I(\mathcal{Y}_m; \mathcal{Z}_m | \mathcal{Z}_{< m})}{\partial p(z_m | y_m)} &= p(y_m) \log p(z_m | y_m) + p(y_m) \frac{p(z_m | y_m)}{p(z_m | y_m)} \\
 &\quad - \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{z}_{< m}) \log p(z_m | \mathbf{z}_{< m}) \\
 &\quad - \sum_{\mathbf{z}_{< m}} \left[ \underbrace{\sum_{y_m} p(z_m | y_m) p(y_m, \mathbf{z}_{< m})}_{=p(\mathbf{z}_{\leq m})} \right] \frac{p(y_m | \mathbf{z}_{< m})}{p(z_m | \mathbf{z}_{< m})} \\
 &= p(y_m) \log p(z_m | y_m) - \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{z}_{< m}) \log p(z_m | \mathbf{z}_{< m}) \\
 &= \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{z}_{< m}) \log \frac{p(z_m | y_m)}{p(z_m | \mathbf{z}_{< m})}. \tag{B.8}
 \end{aligned}$$

### B.3 Fusion of Derived Parts

Combining the result in (B.6) and (B.8) delivers the complete derivative

$$\begin{aligned}
 &- \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{z}_{< m}) \cdot D_{\text{KL}} [p(x | y_m, \mathbf{z}_{< m}) \| p(x | \mathbf{z}_{\leq m})] \\
 &- \beta_m p(y_m) \log p(z_m | y_m) + \beta_m \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{z}_{< m}) \cdot \log p(z_m | \mathbf{z}_{< m}) = 0. \tag{B.9}
 \end{aligned}$$

Following the idea of Blahut and Arimoto [CT06],  $p(x | \mathbf{z})$  and  $p(z_m | \mathbf{z}_{< m})$  are assumed to be independent of  $p(z_m | y_m)$ . With this trick, (B.9) can be resolved w.r.t. the desired mapping of sensor  $m$  leading to the implicit solution

$$p(z_m | y_m) = \frac{e^{-d_{\beta_m}(y_m, z_m)}}{\sum_{z_m} e^{-d_{\beta_m}(y_m, z_m)}} \tag{B.10}$$

with

$$\begin{aligned}
 d_{\beta_m}(y_m, z_m) &:= \sum_{\mathbf{z}_{< m}} p(\mathbf{z}_{< m} | y_m) \left[ \frac{1}{\beta_m} D_{\text{KL}} [p(x | y_m, \mathbf{z}_{< m}) \| p(x | \mathbf{z}_{\leq m})] - \log p(z_m | \mathbf{z}_{< m}) \right] \\
 &= \mathbb{E}_{\mathbf{z}_{< m} | y_m} \left[ \frac{1}{\beta_m} D_{\text{KL}} [p(x | y_m, \mathbf{z}_{< m}) \| p(x | \mathbf{z}_{\leq m})] - \log p(z_m | \mathbf{z}_{< m}) \right]. \tag{B.11}
 \end{aligned}$$

### B.4 Calculating required pmfs

This section covers the calculation of the required pmfs for the previously described algorithm. The first term in the KL divergence in (B.11) is calculated by

$$p(x | y_m, \mathbf{z}_{< m}) = \frac{p(x, y_m, \mathbf{z}_{< m})}{\sum_x p(x, y_m, \mathbf{z}_{< m})} \tag{B.12}$$

where  $p(x, y_m, \mathbf{z}_{<m})$  is determined by

$$p(x, y_m, \mathbf{z}_{<m}) = p(y_m|x)p(\mathbf{z}_{<m}|x)p(x). \quad (\text{B.13})$$

In (B.13) the pmf  $p(y_m|x)$  is given as the measurement channel while  $p(\mathbf{z}_{<m}|x)$  is already calculated by the previous sensor with

$$p(\mathbf{z}_{\leq m}|x) = \sum_{\mathbf{y}_{\leq m}} \prod_{i=1}^m p(z_i|y_i)p(y_i|x) \quad (\text{B.14})$$

where  $p(z_i|y_i)$  is the quantizer mapping of a specific sensor  $i$  in the product. The second term in the KL divergence in (B.11) is calculated by

$$p(x|\mathbf{z}_{\leq m}) = \frac{p(\mathbf{z}_{\leq m}|x)p(x)}{\sum_x p(\mathbf{z}_{\leq m}|x)p(x)}. \quad (\text{B.15})$$

The argument of the logarithm in (B.11) is determined as

$$p(z_m|\mathbf{z}_{<m}) = \frac{\sum_x p(\mathbf{z}_{\leq m}|x)p(x)}{\sum_{z_m} \sum_x p(\mathbf{z}_{\leq m}|x)p(x)}. \quad (\text{B.16})$$

Finally, the pmf to calculate the conditional expectation in (B.11) is given as

$$p(\mathbf{z}_{<m}|y_m) = \frac{\sum_x p(x, y_m, \mathbf{z}_{<m})}{\sum_x p(y_m|x)p(x)} \quad (\text{B.17})$$

where  $p(x, y_m, \mathbf{z}_{<m})$  is already calculated in (B.13). Note that all the above equations simplify when optimizing the first sensor to the scalar IB equations given in Section 3.4.

## Appendix C

### Derivation of an IB-Based Algorithm to Generate Broadcasting Side-Information

As stated in Section 5.2.1, the optimization problem for sensor  $m$  to obtain the instantaneous side-information in a network applying the successive broadcasting protocol is given as

$$L_{\text{BC-SIDE}}^{(m)}[p(s_m|y_m, \mathbf{s}_{<m})] = I(\mathcal{X}; \mathcal{S}_m | \mathcal{S}_{<m}) - \beta_m I(\mathcal{Y}_m; \mathcal{S}_m | \mathcal{S}_{<m}), \quad (\text{C.1})$$

which can be solved by taking the derivative w.r.t. the mapping  $p(s_m|y_m, \mathbf{s}_{<m})$  and equating it to zero. In the following, the derivatives of both mutual information are given.

#### C.1 Derivative of $I(\mathcal{X}; \mathcal{S}_m | \mathcal{S}_{<m})$

The relevant mutual information in (C.1) can be rewritten such that the desired mapping occurs explicitly.

$$\begin{aligned} I(\mathcal{X}; \mathcal{S}_m | \mathcal{S}_{<m}) &= \mathbb{E}_{\mathcal{X}, \mathcal{S}_m, \mathcal{S}_{<m}} \left[ \log \frac{p(s_m|x, \mathbf{s}_{<m})}{p(s_m|\mathbf{s}_{<m})} \right] \\ &= \sum_{s_m} \sum_x \sum_{y_m} \sum_{\mathbf{s}_{<m}} p(s_m|y_m, \mathbf{s}_{<m}) p(y_m, \mathbf{s}_{<m}, x) \\ &\quad \cdot \log \sum_{a \in \mathbb{Y}_m} p(s_m|a, \mathbf{s}_{<m}) p(a|\mathbf{s}_{<m}, x) \\ &- \sum_{s_m} \sum_x \sum_{y_m} \sum_{\mathbf{s}_{<m}} p(s_m|y_m, \mathbf{s}_{<m}) p(y_m, \mathbf{s}_{<m}, x) \\ &\quad \cdot \log \sum_{a \in \mathbb{Y}_m} p(s_m|a, \mathbf{s}_{<m}) p(a|\mathbf{s}_{<m}) \end{aligned} \quad (\text{C.2})$$

The derivative of (C.2) delivers

$$\begin{aligned}
\frac{\partial I(\mathcal{X}; \mathcal{S}_m | \mathcal{S}_{< m})}{\partial p(s_m | y_m, \mathbf{s}_{< m})} &= \sum_x p(x, y_m, \mathbf{s}_{< m}) \cdot \log p(s_m | x, \mathbf{s}_{< m}) \\
&+ \sum_x \left[ \underbrace{\sum_{y_m} p(s_m | y_m, \mathbf{s}_{< m}) \cdot p(x, y_m, \mathbf{s}_{< m})}_{=p(s_m, \mathbf{s}_{< m}, x)} \right] \cdot \frac{p(y_m | x, \mathbf{s}_{< m})}{p(s_m | x, \mathbf{s}_{< m})} \\
&- \sum_x p(x, y_m, \mathbf{s}_{< m}) \cdot \log p(s_m | \mathbf{s}_{< m}) \\
&- \sum_x \left[ \underbrace{\sum_{y_m} p(s_m | y_m, \mathbf{s}_{< m}) \cdot p(x, y_m, \mathbf{s}_{< m})}_{=p(s_m, \mathbf{s}_{< m}, x)} \right] \cdot \frac{p(y_m | \mathbf{s}_{< m})}{p(s_m | \mathbf{s}_{< m})} \\
&= \sum_x p(x, y_m, \mathbf{s}_{< m}) \cdot \log \frac{p(s_m | x, \mathbf{s}_{< m})}{p(s_m | \mathbf{s}_{< m})} \\
&= p(y_m, \mathbf{s}_{< m}) \sum_x p(x | y_m, \mathbf{s}_{< m}) \log \frac{p(s_m | x, \mathbf{s}_{< m})}{p(s_m | \mathbf{s}_{< m})}. \tag{C.3}
\end{aligned}$$

Exploiting Bayes' theorem, the argument of the logarithmic function can be rewritten to

$$\frac{p(s_m | x, \mathbf{s}_{< m})}{p(s_m | \mathbf{s}_{< m})} = \frac{p(x | \mathbf{s}_{\leq m})}{p(x | \mathbf{s}_{< m})} = \frac{p(x | \mathbf{s}_{\leq m})}{p(x | y_m, \mathbf{s}_{< m})} \cdot \frac{p(x | y_m, \mathbf{s}_{< m})}{p(x | \mathbf{s}_{< m})}. \tag{C.4}$$

The last ratio in (C.4) can be dropped because it does not depend on  $p(s_m | y_m, \mathbf{s}_{< m})$  and its contribution can be incorporated into the Lagrange multiplier  $\beta_m$ . The insertion of the first ratio into (C.3) yields the contribution of the derivative of the relevant mutual information

$$\begin{aligned}
\frac{\partial I(\mathcal{X}; \mathcal{S}_m | \mathcal{S}_{< m})}{\partial p(s_m | y_m, \mathbf{s}_{< m})} &\rightarrow -p(y_m, \mathbf{s}_{< m}) \sum_x p(x | y_m, \mathbf{s}_{< m}) \log \frac{p(x | y_m, \mathbf{s}_{< m})}{p(x | \mathbf{s}_{\leq m})} \\
&= -p(y_m, \mathbf{s}_{< m}) \cdot D_{\text{KL}} [p(x | y_m, \mathbf{s}_{< m}) || p(x | \mathbf{s}_{\leq m})]. \tag{C.5}
\end{aligned}$$

## C.2 Derivative of $I(\mathcal{Y}_m; \mathcal{S}_m | \mathcal{S}_{< m})$

With the definition of the conditional compression rate

$$\begin{aligned}
I(\mathcal{Y}_m; \mathcal{S}_m | \mathcal{S}_{< m}) &= \mathbb{E}_{y_m, s_m, \mathbf{s}_{< m}} \left[ \log \frac{p(s_m | y_m, \mathbf{s}_{< m})}{p(s_m | \mathbf{s}_{< m})} \right] \\
&= \sum_{s_m} \sum_{\mathbf{s}_{< m}} \sum_{y_m} p(s_m | y_m, \mathbf{s}_{< m}) p(y_m, \mathbf{s}_{< m}) \log p(s_m | y_m, \mathbf{s}_{< m}) \\
&- \sum_{s_m} \sum_{\mathbf{s}_{< m}} \sum_{y_m} p(s_m | y_m, \mathbf{s}_{< m}) p(y_m, \mathbf{s}_{< m}) \log \sum_{a \in \mathbb{Y}_m} p(s_m | a, \mathbf{s}_{< m}) p(a | \mathbf{s}_{< m}), \tag{C.6}
\end{aligned}$$



its derivative becomes

$$\begin{aligned}
 \frac{\partial I(\mathcal{Y}_m; \mathcal{S}_m | \mathbf{s}_{<m})}{\partial p(s_m | y_m, \mathbf{s}_{<m})} &= p(y_m, \mathbf{s}_{<m}) \log p(s_m | y_m, \mathbf{s}_{<m}) + p(y_m, \mathbf{s}_{<m}) \frac{p(s_m | y_m, \mathbf{s}_{<m})}{p(s_m | y_m, \mathbf{s}_{<m})} \\
 &\quad - p(y_m, \mathbf{s}_{<m}) \log p(s_m | \mathbf{s}_{<m}) \\
 &\quad - \underbrace{\left[ \sum_{y_m} p(s_m | y_m, \mathbf{s}_{<m}) p(y_m, \mathbf{s}_{<m}) \right]}_{=p(s_m, \mathbf{s}_{<m})} \frac{p(y_m | \mathbf{s}_{<m})}{p(s_m | \mathbf{s}_{<m})} \\
 &= p(y_m, \mathbf{s}_{<m}) \log p(s_m | y_m, \mathbf{s}_{<m}) - p(y_m, \mathbf{s}_{<m}) \log p(s_m | \mathbf{s}_{<m}) \\
 &= p(y_m, \mathbf{s}_{<m}) \log \frac{p(s_m | y_m, \mathbf{s}_{<m})}{p(s_m | \mathbf{s}_{<m})}. \tag{C.7}
 \end{aligned}$$

### C.3 Fusion of Derived Parts

Combining the result in (C.5) and (C.7) delivers the complete derivative

$$\begin{aligned}
 &- p(y_m, \mathbf{s}_{<m}) \cdot D_{\text{KL}} [p(x | y_m, \mathbf{s}_{<m}) || p(x | \mathbf{s}_{<m})] \\
 &- \beta_m p(y_m, \mathbf{s}_{<m}) \log p(s_m | y_m, \mathbf{s}_{<m}) + \beta_m p(y_m, \mathbf{s}_{<m}) \cdot \log p(s_m | \mathbf{s}_{<m}) = 0. \tag{C.8}
 \end{aligned}$$

Following the idea of Blahut and Arimoto [CT06],  $p(x | \mathbf{s}_{<m})$  and  $p(s_m | \mathbf{s}_{<m})$  are assumed to be independent of  $p(s_m | y_m, \mathbf{s}_{<m})$ . With this trick, (C.8) can be resolved w.r.t. the desired mapping of sensor  $m$  leading to the implicit solution

$$p(s_m | y_m, \mathbf{s}_{<m}) = \frac{e^{-d_{\beta_m}(y_m, s_m, \mathbf{s}_{<m})}}{\sum_{s_m} e^{-d_{\beta_m}(y_m, s_m, \mathbf{s}_{<m})}} \tag{C.9}$$

with

$$d_{\beta_m}(y_m, s_m, \mathbf{s}_{<m}) := \frac{1}{\beta_m} D_{\text{KL}} [p(x | y_m, \mathbf{s}_{<m}) || p(x | \mathbf{s}_{<m})] - \log p(s_m | \mathbf{s}_{<m}). \tag{C.10}$$

### C.4 Calculating required pmfs

This section covers the calculation of the required pmfs for the previously described algorithm. The first term in the KL divergence in (C.10) is calculated by

$$p(x | y_m, \mathbf{s}_{<m}) = \frac{p(y_m | x) p(\mathbf{s}_{<m} | x) p(x)}{p(y_m, \mathbf{s}_{<m})} \tag{C.11}$$

where  $p(y_m | x)$  is given as the measuring channel and  $p(\mathbf{s}_{<m} | x)$  is calculated recursively by the previous sensor by

$$p(\mathbf{s}_{<m} | x) = \sum_{y_m} p(s_m | y_m, \mathbf{s}_{<m}) p(y_m | x) p(\mathbf{s}_{<m} | x). \tag{C.12}$$

In (C.12) the pmf  $p(s_m | y_m, \mathbf{s}_{<m})$  denotes the current mapping of the instantaneous side-information and  $p(\mathbf{s}_{<m} | x)$  is a recursive term, which can already be computed by the

previous sensor  $m - 1$ . The pmf  $p(y_m, \mathbf{s}_{<m})$  in the denominator of equation (C.11) can be calculated as

$$p(y_m, \mathbf{s}_{<m}) = \sum_x p(y_m|x)p(\mathbf{s}_{<m}|x)p(x) . \quad (\text{C.13})$$

The second term of the KL divergence in (C.10) can be calculated as

$$p(x|\mathbf{s}_{\leq m}) = \frac{p(\mathbf{s}_{\leq m}|x)p(x)}{\sum_x p(\mathbf{s}_{\leq m}|x)p(x)} \quad (\text{C.14})$$

where  $p(\mathbf{s}_{\leq m}|x)$  is already calculated in (C.12). Finally, the term in the argument of the logarithm in (C.10) can be calculated as

$$p(s_m|\mathbf{s}_{<m}) = \frac{\sum_x p(\mathbf{s}_{\leq m}|x)p(x)}{\sum_x \sum_{s_m} p(\mathbf{s}_{\leq m}|x)p(x)} . \quad (\text{C.15})$$

Note that all the above equations simplify when optimizing the first sensor to the scalar IB equations given in Section 3.4.

## Appendix D

### Derivation of a GDIB Algorithm for Cooperating Sensors Applying the Successive Broadcasting Protocol

For extending the GDIB algorithm in order to allow the sensors to exchange instantaneous side-information with the successive broadcasting protocol, the optimization problem given in Subsection 5.2.2

$$L_{\text{GDIB-BC}}^{(m)}[p(z_m|y_m, \mathbf{s}_{<m})] = I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{<m}) - \beta_m I(\mathcal{Y}_m, \mathcal{S}_{<m}; \mathcal{Z}_m | \mathcal{Z}_{<m}) \quad (\text{D.1})$$

has to be solved by taking the derivative w.r.t. the mapping  $p(z_m|y_m, \mathbf{s}_{<m})$  and equating it to zero. In the following, the derivatives of both mutual information are given.

#### D.1 Derivative of $I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{<m})$

The relevant mutual information in (D.1) can be rewritten such that the desired mapping occurs explicitly.

$$\begin{aligned} I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{<m}) &= \mathbb{E}_{\mathcal{X}, \mathcal{Z}_m, \mathcal{Z}_{<m}} \left[ \log \frac{p(z_m|x, \mathbf{z}_{<m})}{p(z_m|\mathbf{z}_{<m})} \right] \\ &= \sum_{z_m} \sum_{\mathbf{z}_{<m}} \sum_x \sum_{y_m} \sum_{\mathbf{s}_{<m}} p(z_m|y_m, \mathbf{s}_{<m}) p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}, x) \\ &\quad \cdot \log \sum_{a \in \mathcal{Y}_m} \sum_{b \in \mathcal{S}_{<m}} p(z_m|a, b) p(a, b | \mathbf{z}_{<m}, x) \\ &- \sum_{z_m} \sum_{\mathbf{z}_{<m}} \sum_x \sum_{y_m} \sum_{\mathbf{s}_{<m}} p(z_m|y_m, \mathbf{s}_{<m}) p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}, x) \\ &\quad \cdot \log \sum_{a \in \mathcal{Y}_m} \sum_{b \in \mathcal{S}_{<m}} p(z_m|a, b) p(a, b | \mathbf{z}_{<m}) \end{aligned} \quad (\text{D.2})$$

The derivative of (D.2) delivers

$$\begin{aligned}
\frac{\partial I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{< m})}{\partial p(z_m | y_m, \mathbf{s}_{< m})} &= \sum_x \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m}, x) \cdot \log p(z_m | x, \mathbf{z}_{< m}) \\
&+ \sum_x \sum_{\mathbf{z}_{< m}} \underbrace{\left[ \sum_{\mathbf{s}_{< m}} \sum_{y_m} p(z_m | y_m, \mathbf{s}_{< m}) \cdot p(y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m}, x) \right]}_{=p(z_m, \mathbf{z}_{< m}, x)} \cdot \frac{p(y_m, \mathbf{s}_{< m} | x, \mathbf{z}_{< m})}{p(z_m | x, \mathbf{z}_{< m})} \\
&- \sum_x \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m}, x) \cdot \log p(z_m | \mathbf{z}_{< m}) \\
&- \sum_x \sum_{\mathbf{z}_{< m}} \underbrace{\left[ \sum_{\mathbf{s}_{< m}} \sum_{y_m} p(z_m | y_m, \mathbf{s}_{< m}) \cdot p(y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m}, x) \right]}_{=p(z_m, \mathbf{z}_{< m}, x)} \cdot \frac{p(y_m, \mathbf{s}_{< m} | \mathbf{z}_{< m})}{p(z_m | \mathbf{z}_{< m})} \\
&= \sum_x \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m}, x) \cdot \log \frac{p(z_m | x, \mathbf{z}_{< m})}{p(z_m | \mathbf{z}_{< m})} \\
&= \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m}) \sum_x p(x | y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m}) \log \frac{p(z_m | x, \mathbf{z}_{< m})}{p(z_m | \mathbf{z}_{< m})}. \tag{D.3}
\end{aligned}$$

Exploiting Bayes' theorem, the argument of the logarithmic function can be rewritten to

$$\frac{p(z_m | x, \mathbf{z}_{< m})}{p(z_m | \mathbf{z}_{< m})} = \frac{p(x | \mathbf{z}_{\leq m})}{p(x | \mathbf{z}_{< m})} = \frac{p(x | \mathbf{z}_{\leq m})}{p(x | y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m})} \cdot \frac{p(x | y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m})}{p(x | \mathbf{z}_{\leq m})}. \tag{D.4}$$

The last ratio in (D.4) can be dropped because it does not depend on  $p(z_m | y_m, \mathbf{s}_{< m})$  and its contribution can be incorporated into the Lagrange multiplier  $\beta_m$ . The insertion of the first ratio into (D.3) yields the contribution of the derivative of the relevant mutual information

$$\begin{aligned}
\frac{\partial I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{< m})}{\partial p(z_m | y_m, \mathbf{s}_{< m})} &\rightarrow - \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m}) \sum_x p(x | y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m}) \log \frac{p(x | y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m})}{p(x | \mathbf{z}_{\leq m})} \\
&= - \sum_{\mathbf{z}_{< m}} p(y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m}) \cdot D_{\text{KL}} [p(x | y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m}) || p(x | \mathbf{z}_{\leq m})]. \tag{D.5}
\end{aligned}$$

## D.2 Derivative of $I(\mathcal{Y}_m, \mathcal{S}_{< m}; \mathcal{Z}_m | \mathcal{Z}_{< m})$

With the definition of the conditional compression rate

$$\begin{aligned}
I(\mathcal{Y}_m, \mathcal{S}_{< m}; \mathcal{Z}_m | \mathcal{Z}_{< m}) &= \mathbb{E}_{\mathcal{Y}_m, \mathcal{Z}_m, \mathcal{Z}_{< m}, \mathcal{S}_{< m}} \left[ \log \frac{p(z_m | y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m})}{p(z_m | \mathbf{z}_{< m})} \right] \\
&= \sum_{z_m} \sum_{\mathbf{z}_{< m}} \sum_{\mathbf{s}_{< m}} \sum_{y_m} p(z_m | y_m, \mathbf{s}_{< m}) p(y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m}) \log p(z_m | y_m, \mathbf{s}_{< m}) \\
&- \sum_{z_m} \sum_{\mathbf{z}_{< m}} \sum_{\mathbf{s}_{< m}} \sum_{y_m} p(z_m | y_m, \mathbf{s}_{< m}) p(y_m, \mathbf{s}_{< m}, \mathbf{z}_{< m}) \log \sum_{a \in \mathbb{Y}_m} \sum_{b \in \mathbb{S}_{< m}} p(z_m | a, b) p(a, b | \mathbf{z}_{< m})
\end{aligned} \tag{D.6}$$

its derivative becomes

$$\begin{aligned}
& \frac{\partial I(\mathcal{Y}_m, \mathbf{s}_{<m}; \mathcal{Z}_m | \mathcal{Z}_{<m})}{\partial p(z_m | y_m, \mathbf{s}_{<m})} \\
&= \sum_{\mathbf{z}_{<m}} p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) \log p(z_m | y_m, \mathbf{s}_{<m}) \\
&+ \sum_{\mathbf{z}_{<m}} p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) \frac{p(z_m | y_m, \mathbf{s}_{<m})}{p(z_m | y_m, \mathbf{s}_{<m})} \\
&- \sum_{\mathbf{z}_{<m}} p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) \log p(z_m | \mathbf{z}_{<m}) \\
&- \sum_{\mathbf{z}_{<m}} \underbrace{\left[ \sum_{y_m} \sum_{\mathbf{s}_{<m}} p(z_m | y_m, \mathbf{s}_{<m}) p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) \right]}_{=p(z_m, \mathbf{z}_{<m})} \frac{p(y_m, \mathbf{s}_{<m} | \mathbf{z}_{<m})}{p(z_m | \mathbf{z}_{<m})} \\
&= \sum_{\mathbf{z}_{<m}} p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) \log p(z_m | y_m, \mathbf{s}_{<m}) \\
&- \sum_{\mathbf{z}_{<m}} p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) \log p(z_m | \mathbf{z}_{<m}) \\
&= \sum_{\mathbf{z}_{<m}} p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) \log \frac{p(z_m | y_m, \mathbf{s}_{<m})}{p(z_m | \mathbf{z}_{<m})}. \tag{D.7}
\end{aligned}$$

### D.3 Fusion of Derived Parts

Combining the result in (D.5) and (D.7) delivers the complete derivative

$$\begin{aligned}
& - \sum_{\mathbf{z}_{<m}} p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) \cdot D_{\text{KL}} [p(x | y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) || p(x | \mathbf{z}_{\leq m})] \\
& - \beta_m \sum_{\mathbf{z}_{<m}} p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) \log p(z_m | y_m, \mathbf{s}_{<m}) \\
& + \beta_m \sum_{\mathbf{z}_{<m}} p(y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) \cdot \log p(z_m | \mathbf{z}_{<m}) = 0. \tag{D.8}
\end{aligned}$$

Following the idea of Blahut and Arimoto [CT06],  $p(x | \mathbf{z}_{\leq m})$  and  $p(z_m | \mathbf{z}_{<m})$  are assumed to be independent of  $p(z_m | y_m, \mathbf{s}_{<m})$ . With this trick, (D.8) can be resolved w.r.t. the desired mapping of sensor  $m$  leading to the implicit solution

$$p(z_m | y_m, \mathbf{s}_{<m}) = \frac{e^{-d_{\beta_m}(y_m, z_m, \mathbf{s}_{<m})}}{\sum_{z_m} e^{-d_{\beta_m}(y_m, z_m, \mathbf{s}_{<m})}} \tag{D.9}$$

with

$$\begin{aligned}
& d_{\beta_m}(y_m, z_m, \mathbf{s}_{<m}) \\
& := \sum_{\mathbf{z}_{<m}} p(\mathbf{z}_{<m} | y_m, \mathbf{s}_{<m}) \left[ \frac{1}{\beta_m} D_{\text{KL}} [p(x | y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) || p(x | \mathbf{z}_{\leq m})] - \log p(z_m | \mathbf{z}_{<m}) \right] \\
& = \mathbb{E}_{\mathbf{Z}_{<m} | y_m, \mathbf{s}_{<m}} \left[ \frac{1}{\beta_m} D_{\text{KL}} [p(x | y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) || p(x | \mathbf{z}_{\leq m})] - \log p(z_m | \mathbf{z}_{<m}) \right]. \tag{D.10}
\end{aligned}$$

## D.4 Calculating required pmfs

This section covers the calculation of the required pmfs for the previously described algorithm. The first term in the KL divergence in (D.10) is calculated by

$$p(x|y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) = \frac{p(x, y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m})}{\sum_x p(x, y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m})} \quad (\text{D.11})$$

where  $p(x, y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m})$  is determined recursively as

$$p(x, y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) = \sum_{y_{m-1}} p(z_{m-1}|y_{m-1}, \mathbf{s}_{<m-1}) p(s_{m-1}|y_{m-1}, \mathbf{s}_{<m-1}) p(y_m|x) p(x, y_{m-1}, \mathbf{s}_{<m-1}, \mathbf{z}_{<m-1}). \quad (\text{D.12})$$

In (D.12) the pmf  $p(z_{m-1}|y_{m-1}, \mathbf{s}_{<m-1})$  is given as the quantizer mapping of the previous sensor while  $p(s_{m-1}|y_{m-1}, \mathbf{s}_{<m-1})$  is given as the mapping for the instantaneous side-information of the previous sensor. The recursive pmf  $p(x, y_{m-1}, \mathbf{s}_{<m-1}, \mathbf{z}_{<m-1})$  can already be calculated by previous sensors. The second term in the KL divergence in (D.10) is calculated by

$$p(x|\mathbf{z}_{\leq m}) = \frac{p(\mathbf{z}_{\leq m}, x)}{\sum_x p(\mathbf{z}_{\leq m}, x)} \quad (\text{D.13})$$

where  $p(\mathbf{z}_{\leq m}, x)$  can be calculated by

$$p(\mathbf{z}_{\leq m}, x) = \sum_{\mathbf{s}_{<m}} \sum_{y_m} p(z_m|y_m, \mathbf{s}_{<m}) p(x, y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m}) \quad (\text{D.14})$$

with  $p(z_m|y_m, \mathbf{s}_{<m})$  being the quantizer mapping of the current sensor and  $p(x, y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m})$  being already calculated in (D.12). The argument of the logarithm in (D.10) can be derived as

$$p(z_m|\mathbf{z}_{<m}) = \frac{\sum_x p(\mathbf{z}_{\leq m}, x)}{\sum_{z_m} \sum_x p(\mathbf{z}_{\leq m}, x)}. \quad (\text{D.15})$$

Finally, the pmf to calculate the conditional expectation in (D.10) is determined as

$$p(\mathbf{z}_{<m}|y_m, \mathbf{s}_{<m}) = \frac{\sum_x p(x, y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m})}{\sum_{z_{<m}} \sum_x p(x, y_m, \mathbf{s}_{<m}, \mathbf{z}_{<m})}. \quad (\text{D.16})$$

Note that all the above equations simplify when optimizing the first sensor to the scalar IB equations given in Section 3.4.

## Appendix E

### Derivation of an IB-Based Algorithm to Generate Point-To-Point Side-Information

As stated in Subsection 5.3.1, the optimization problem for sensor  $m$  to generate the instantaneous side-information in a network with cooperating sensors applying the sequential point-to-point protocol is given as

$$L_{\text{PTP-SIDE}}^{(m)}[p(s_m|y_m, s_{m-1})] = I(\mathcal{X}; \mathcal{S}_m) - \beta_m I(\mathcal{Y}_m, \mathcal{S}_{m-1}; \mathcal{S}_m), \quad (\text{E.1})$$

which can be solved by taking the derivative w.r.t. the mapping  $p(s_m|y_m, s_{m-1})$  and equating it to zero. In the following, the derivatives of both mutual information are given.

#### E.1 Derivative of $I(\mathcal{X}; \mathcal{S}_m)$

The relevant mutual information in (E.1) can be rewritten such that the desired mapping occurs explicitly.

$$\begin{aligned} I(\mathcal{X}; \mathcal{S}_m) &= \mathbb{E}_{\mathcal{X}, \mathcal{S}_m} \left[ \log \frac{p(s_m|x)}{p(s_m)} \right] \\ &= \sum_{s_m} \sum_x \sum_{y_m} \sum_{s_{m-1}} p(s_m|y_m, s_{m-1}) p(y_m, s_{m-1}, x) \\ &\quad \cdot \log \sum_{a \in \mathbb{Y}_m} \sum_{b \in \mathbb{S}_{m-1}} p(s_m|a, b) p(a, b|x) \\ &\quad - \sum_{s_m} \sum_x \sum_{y_m} \sum_{s_{m-1}} p(s_m|y_m, s_{m-1}) p(y_m, s_{m-1}, x) \\ &\quad \cdot \log \sum_{a \in \mathbb{Y}_m} \sum_{b \in \mathbb{S}_{m-1}} p(s_m|a, b) p(a, b) \end{aligned} \quad (\text{E.2})$$

The derivative of (E.2) delivers

$$\begin{aligned}
\frac{\partial I(\mathcal{X}; \mathcal{S}_m)}{\partial p(s_m|y_m, s_{m-1})} &= \sum_x p(x, y_m, s_{m-1}) \cdot \log p(s_m|x) \\
&+ \sum_x \left[ \underbrace{\sum_{s_{m-1}} \sum_{y_m} p(s_m|y_m, s_{m-1}) \cdot p(x, y_m, s_{m-1})}_{=p(s_m, x)} \right] \cdot \frac{p(y_m, s_{m-1}|x)}{p(s_m|x)} \\
&- \sum_x p(x, y_m, s_{m-1}) \cdot \log p(s_m) \\
&- \sum_x \left[ \underbrace{\sum_{s_{m-1}} \sum_{y_m} p(s_m|y_m, s_{m-1}) \cdot p(x, y_m, s_{m-1})}_{=p(s_m, x)} \right] \cdot \frac{p(y_m, s_{m-1})}{p(s_m)} \\
&= \sum_x p(x, y_m, s_{m-1}) \cdot \log \frac{p(s_m|x)}{p(s_m)} \\
&= p(y_m, s_{m-1}) \sum_x p(x|y_m, s_{m-1}) \log \frac{p(s_m|x)}{p(s_m)}. \tag{E.3}
\end{aligned}$$

Exploiting Bayes' theorem, the argument of the logarithmic function can be rewritten to

$$\frac{p(s_m|x)}{p(s_m)} = \frac{p(x|s_m)}{p(x)} = \frac{p(x|s_m)}{p(x|y_m, s_{m-1})} \cdot \frac{p(x|y_m, s_{m-1})}{p(x)}. \tag{E.4}$$

The last ratio in (E.4) can be dropped because it does not depend on  $p(s_m|y_m, s_{m-1})$  and its contribution can be incorporated into the Lagrange multiplier  $\beta_m$ . The insertion of the first ratio into (E.3) yields the contribution of the derivative of the relevant mutual information

$$\begin{aligned}
\frac{\partial I(\mathcal{X}; \mathcal{S}_m)}{\partial p(s_m|y_m, s_{m-1})} &\rightarrow -p(y_m, s_{m-1}) \sum_x p(x|y_m, s_{m-1}) \log \frac{p(x|y_m, s_{m-1})}{p(x|s_m)} \\
&= -p(y_m, s_{m-1}) \cdot D_{\text{KL}} [p(x|y_m, s_{m-1}) || p(x|s_m)]. \tag{E.5}
\end{aligned}$$

## E.2 Derivative of $I(\mathcal{Y}_m, \mathcal{S}_{m-1}; \mathcal{S}_m)$

With the definition of the conditional compression rate

$$\begin{aligned}
I(\mathcal{Y}_m, \mathcal{S}_{m-1}; \mathcal{S}_m) &= \mathbb{E}_{y_m, s_m, s_{m-1}} \left[ \log \frac{p(s_m|y_m, s_{m-1})}{p(s_m)} \right] \\
&= \sum_{s_m} \sum_{s_{m-1}} \sum_{y_m} p(s_m|y_m, s_{m-1}) p(y_m, s_{m-1}) \log p(s_m|y_m, s_{m-1}) \\
&- \sum_{s_m} \sum_{s_{m-1}} \sum_{y_m} p(s_m|y_m, s_{m-1}) p(y_m, s_{m-1}) \log \sum_{a \in \mathcal{Y}_m} \sum_{b \in \mathcal{S}_{m-1}} p(s_m|a, b) p(a, b), \tag{E.6}
\end{aligned}$$



its derivative becomes

$$\begin{aligned}
 \frac{\partial I(\mathcal{Y}_m; \mathcal{S}_{m-1}; \mathcal{S}_m)}{\partial p(s_m|y_m, s_{m-1})} &= p(y_m, s_{m-1}) \log p(s_m|y_m, s_{m-1}) + p(y_m, s_{m-1}) \frac{p(s_m|y_m, s_{m-1})}{p(s_m|y_m, s_{m-1})} \\
 &\quad - p(y_m, s_{m-1}) \log p(s_m) \\
 &\quad - \underbrace{\left[ \sum_{y_m} \sum_{s_{m-1}} p(s_m|y_m, s_{m-1}) p(y_m, s_{m-1}) \right]}_{=p(s_m)} \frac{p(y_m, s_{m-1})}{p(s_m)} \\
 &= p(y_m, s_{m-1}) \log p(s_m|y_m, s_{m-1}) - p(y_m, s_{m-1}) \log p(s_m) \\
 &= p(y_m, s_{m-1}) \log \frac{p(s_m|y_m, s_{m-1})}{p(s_m)}. \tag{E.7}
 \end{aligned}$$

### E.3 Fusion of Derived Parts

Combining the result in (E.5) and (E.7) delivers the complete derivative

$$\begin{aligned}
 &- p(y_m, s_{m-1}) \cdot D_{\text{KL}} [p(x|y_m, s_{m-1}) || p(x|s_m)] \\
 &- \beta_m p(y_m, s_{m-1}) \log p(s_m|y_m, s_{m-1}) + \beta_m p(y_m, s_{m-1}) \cdot \log p(s_m) = 0. \tag{E.8}
 \end{aligned}$$

Following the idea of Blahut and Arimoto [CT06],  $p(x|s_m)$  and  $p(s_m)$  are assumed to be independent of  $p(s_m|y_m, s_{m-1})$ . With this trick, (E.8) can be resolved w.r.t. the desired mapping of sensor  $m$  leading to the implicit solution

$$p(s_m|y_m, s_{m-1}) = \frac{e^{-d_{\beta_m}(y_m, s_m, s_{m-1})}}{\sum_{s_m} e^{-d_{\beta_m}(y_m, s_m, s_{m-1})}} \tag{E.9}$$

with

$$d_{\beta_m}(y_m, s_m, s_{m-1}) := \frac{1}{\beta_m} D_{\text{KL}} [p(x|y_m, s_{m-1}) || p(x|s_m)] - \log p(s_m). \tag{E.10}$$

### E.4 Calculating required pmfs

This section covers the calculation of the required pmfs for the previously described algorithm. The first term in the KL divergence in (E.10) is calculated by

$$p(x|y_m, s_{m-1}) = \frac{p(y_m|x)p(s_{m-1}|x)p(x)}{\sum_x p(y_m|x)p(s_{m-1}|x)p(x)} \tag{E.11}$$

where  $p(y_m|x)$  is given as the measuring channel,  $p(s_{m-1}|x)$  has already been calculated recursively by the previous sensor with

$$p(s_m|x) = \sum_{y_m} \sum_{s_{m-1}} p(s_m|y_m, s_{m-1}) p(y_m|x) p(s_{m-1}|x). \tag{E.12}$$

In (E.12) the pmf  $p(s_m|y_m, s_{m-1})$  represents the quantizer mapping of the current sensor and  $p(s_{m-1}|x)$  can be determined recursively by previous sensors. The second term in the KL divergence in (E.10) is calculated by

$$p(x|s_m) = \frac{p(s_m|x)p(x)}{\sum_x p(s_m|x)p(x)} \quad (\text{E.13})$$

where  $p(s_m|x)$  is already calculated in (E.12). Finally, the argument in the logarithm in (E.10) can be determined as

$$p(s_m) = \sum_x p(s_m|x)p(x) \quad (\text{E.14})$$

Note that all the above equations simplify when optimizing the first sensor to the scalar IB equations given in Section 3.4.

## Appendix F

### Derivation of a GDIB Algorithm for Cooperating Sensors Applying the Sequential Point-To-Point Protocol

For extending the GDIB algorithm in order to allow the sensors to sequentially communicate with each other, the optimization problem for sensor  $m$  given in Subsection 5.3.2

$$L_{\text{GDIB-PTP}}^{(m)}[p(z_m|y_m, s_{m-1})] = I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{<m}) - \beta_m I(\mathcal{Y}_m, \mathcal{S}_{m-1}; \mathcal{Z}_m | \mathcal{Z}_{<m}) \quad (\text{F.1})$$

has to be solved by taking the derivative w.r.t. the mapping  $p(z_m|y_m, s_{m-1})$  and equating it to zero. In the following, the derivatives of both mutual information are given.

#### F.1 Derivative of $I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{<m})$

The relevant mutual information in (F.1) can be rewritten such that the desired mapping occurs explicitly.

$$\begin{aligned} I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{<m}) &= \mathbb{E}_{\mathcal{X}, \mathcal{Z}_m, \mathcal{Z}_{<m}} \left[ \log \frac{p(z_m|x, \mathbf{z}_{<m})}{p(z_m|\mathbf{z}_{<m})} \right] \\ &= \sum_{z_m} \sum_{\mathbf{z}_{<m}} \sum_x \sum_{y_m} \sum_{s_{m-1}} p(z_m|y_m, s_{m-1}) p(y_m, s_{m-1}, \mathbf{z}_{<m}, x) \\ &\quad \cdot \log \sum_{a \in \mathbb{Y}_m} \sum_{b \in \mathbb{S}_{m-1}} p(z_m|a, b) p(a, b | \mathbf{z}_{<m}, x) \\ &- \sum_{z_m} \sum_{\mathbf{z}_{<m}} \sum_x \sum_{y_m} \sum_{s_{m-1}} p(z_m|y_m, s_{m-1}) p(y_m, s_{m-1}, \mathbf{z}_{<m}, x) \\ &\quad \cdot \log \sum_{a \in \mathbb{Y}_m} \sum_{b \in \mathbb{S}_{m-1}} p(z_m|a, b) p(a, b | \mathbf{z}_{<m}) \end{aligned} \quad (\text{F.2})$$

The derivative of (F.2) delivers

$$\begin{aligned}
\frac{\partial I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{< m})}{\partial p(z_m | y_m, s_{m-1})} &= \sum_x \sum_{\mathbf{z}_{< m}} p(y_m, s_{m-1}, \mathbf{z}_{< m}, x) \cdot \log p(z_m | x, \mathbf{z}_{< m}) \\
&+ \sum_x \sum_{\mathbf{z}_{< m}} \left[ \underbrace{\sum_{s_{m-1}} \sum_{y_m} p(z_m | y_m, s_{m-1}) \cdot p(y_m, s_{m-1}, \mathbf{z}_{< m}, x)}_{=p(z_m, \mathbf{z}_{< m}, x)} \right] \cdot \frac{p(y_m, s_{m-1} | x, \mathbf{z}_{< m})}{p(z_m | x, \mathbf{z}_{< m})} \\
&- \sum_x \sum_{\mathbf{z}_{< m}} p(y_m, s_{m-1}, \mathbf{z}_{< m}, x) \cdot \log p(z_m | \mathbf{z}_{< m}) \\
&- \sum_x \sum_{\mathbf{z}_{< m}} \left[ \underbrace{\sum_{s_{m-1}} \sum_{y_m} p(z_m | y_m, s_{m-1}) \cdot p(y_m, s_{m-1}, \mathbf{z}_{< m}, x)}_{=p(z_m, \mathbf{z}_{< m}, x)} \right] \cdot \frac{p(y_m, s_{m-1} | \mathbf{z}_{< m})}{p(z_m | \mathbf{z}_{< m})} \\
&= \sum_x \sum_{\mathbf{z}_{< m}} p(y_m, s_{m-1}, \mathbf{z}_{< m}, x) \cdot \log \frac{p(z_m | x, \mathbf{z}_{< m})}{p(z_m | \mathbf{z}_{< m})} \\
&= \sum_{\mathbf{z}_{< m}} p(y_m, s_{m-1}, \mathbf{z}_{< m}) \sum_x p(x | y_m, s_{m-1}, \mathbf{z}_{< m}) \log \frac{p(z_m | x, \mathbf{z}_{< m})}{p(z_m | \mathbf{z}_{< m})}. \tag{F.3}
\end{aligned}$$

Exploiting Bayes' theorem, the argument of the logarithmic function can be rewritten to

$$\frac{p(z_m | x, \mathbf{z}_{< m})}{p(z_m | \mathbf{z}_{< m})} = \frac{p(x | \mathbf{z}_{\leq m})}{p(x | \mathbf{z}_{< m})} = \frac{p(x | \mathbf{z}_{\leq m})}{p(x | y_m, s_{m-1}, \mathbf{z}_{< m})} \cdot \frac{p(x | y_m, s_{m-1}, \mathbf{z}_{< m})}{p(x | \mathbf{z}_{< m})}. \tag{F.4}$$

The last ratio in (F.4) can be dropped because it does not depend on  $p(z_m | y_m, s_{m-1})$  and its contribution can be incorporated into the Lagrange multiplier  $\beta_m$ . The insertion of the first ratio into (F.3) yields the contribution of the derivative of the relevant mutual information

$$\begin{aligned}
\frac{\partial I(\mathcal{X}; \mathcal{Z}_m | \mathcal{Z}_{< m})}{\partial p(z_m | y_m, s_{m-1})} &\rightarrow - \sum_{\mathbf{z}_{< m}} p(y_m, s_{m-1}, \mathbf{z}_{< m}) \sum_x p(x | y_m, s_{m-1}, \mathbf{z}_{< m}) \log \frac{p(x | y_m, s_{m-1}, \mathbf{z}_{< m})}{p(x | \mathbf{z}_{\leq m})} \\
&= - \sum_{\mathbf{z}_{< m}} p(y_m, s_{m-1}, \mathbf{z}_{< m}) \cdot D_{\text{KL}} [p(x | y_m, s_{m-1}, \mathbf{z}_{< m}) || p(x | \mathbf{z}_{\leq m})]. \tag{F.5}
\end{aligned}$$

## F.2 Derivative of $I(\mathcal{Y}_m, \mathcal{S}_{m-1}; \mathcal{Z}_m | \mathcal{Z}_{< m})$

With the definition of the conditional compression rate

$$\begin{aligned}
I(\mathcal{Y}_m, \mathcal{S}_{m-1}; \mathcal{Z}_m | \mathcal{Z}_{< m}) &= \mathbb{E}_{\mathcal{Y}_m, \mathcal{Z}_m, \mathcal{Z}_{< m}, \mathcal{S}_{m-1}} \left[ \log \frac{p(z_m | y_m, s_{m-1}, \mathbf{z}_{< m})}{p(z_m | \mathbf{z}_{< m})} \right] \\
&= \sum_{z_m} \sum_{\mathbf{z}_{< m}} \sum_{s_{m-1}} \sum_{y_m} p(z_m | y_m, s_{m-1}) p(y_m, s_{m-1}, \mathbf{z}_{< m}) \log p(z_m | y_m, s_{m-1}) \\
&- \sum_{z_m} \sum_{\mathbf{z}_{< m}} \sum_{s_{m-1}} \sum_{y_m} p(z_m | y_m, s_{m-1}) p(y_m, s_{m-1}, \mathbf{z}_{< m}) \log \sum_{a \in \mathbb{Y}_m} \sum_{b \in \mathbb{S}_{m-1}} p(z_m | a, b) p(a, b | \mathbf{z}_{< m}) \tag{F.6}
\end{aligned}$$

its derivative becomes

$$\begin{aligned}
& \frac{\partial I(\mathcal{Y}_m, \mathcal{S}_{m-1}; \mathcal{Z}_m | \mathcal{Z}_{<m})}{\partial p(z_m | y_m, s_{m-1})} \\
&= \sum_{\mathbf{z}_{<m}} p(y_m, s_{m-1}, \mathbf{z}_{<m}) \log p(z_m | y_m, s_{m-1}) \\
&+ \sum_{\mathbf{z}_{<m}} p(y_m, s_{m-1}, \mathbf{z}_{<m}) \frac{p(z_m | y_m, s_{m-1})}{p(z_m | y_m, s_{m-1})} \\
&- \sum_{\mathbf{z}_{<m}} p(y_m, s_{m-1}, \mathbf{z}_{<m}) \log p(z_m | \mathbf{z}_{<m}) \\
&- \sum_{\mathbf{z}_{<m}} \left[ \underbrace{\sum_{y_m} \sum_{s_{m-1}} p(z_m | y_m, s_{m-1}) p(y_m, s_{m-1}, \mathbf{z}_{<m})}_{=p(z_m, \mathbf{z}_{<m})} \right] \frac{p(y_m, s_{m-1} | \mathbf{z}_{<m})}{p(z_m | \mathbf{z}_{<m})} \\
&= \sum_{\mathbf{z}_{<m}} p(y_m, s_{m-1}, \mathbf{z}_{<m}) \log p(z_m | y_m, s_{m-1}) \\
&- \sum_{\mathbf{z}_{<m}} p(y_m, s_{m-1}, \mathbf{z}_{<m}) \log p(z_m | \mathbf{z}_{<m}) \\
&= \sum_{\mathbf{z}_{<m}} p(y_m, s_{m-1}, \mathbf{z}_{<m}) \log \frac{p(z_m | y_m, s_{m-1})}{p(z_m | \mathbf{z}_{<m})}. \tag{F.7}
\end{aligned}$$

### F.3 Fusion of Derived Parts

Combining the result in (F.5) and (F.7) delivers the complete derivative

$$\begin{aligned}
& - \sum_{\mathbf{z}_{<m}} p(y_m, s_{m-1}, \mathbf{z}_{<m}) \cdot D_{\text{KL}} [p(x | y_m, s_{m-1}, \mathbf{z}_{<m}) || p(x | \mathbf{z}_{\leq m})] \\
& - \beta_m \sum_{\mathbf{z}_{<m}} p(y_m, s_{m-1}, \mathbf{z}_{<m}) \log p(z_m | y_m, s_{m-1}) \\
& + \beta_m \sum_{\mathbf{z}_{<m}} p(y_m, s_{m-1}, \mathbf{z}_{<m}) \cdot \log p(z_m | \mathbf{z}_{<m}) = 0. \tag{F.8}
\end{aligned}$$

Following the idea of Blahut and Arimoto [CT06],  $p(x | \mathbf{z}_{\leq m})$  and  $p(z_m | \mathbf{z}_{<m})$  are assumed to be independent of  $p(z_m | y_m, s_{m-1})$ . With this trick, (F.8) can be resolved w.r.t. the desired mapping of sensor  $m$  leading to the implicit solution

$$p(z_m | y_m, s_{m-1}) = \frac{e^{-d_{\beta_m}(y_m, z_m, s_{m-1})}}{\sum_{z_m} e^{-d_{\beta_m}(y_m, z_m, s_{m-1})}} \tag{F.9}$$

with

$$\begin{aligned}
& d_{\beta_m}(y_m, z_m, s_{m-1}) \\
& := \sum_{\mathbf{z}_{<m}} p(\mathbf{z}_{<m} | y_m, s_{m-1}) \left[ \frac{1}{\beta_m} D_{\text{KL}} [p(x | y_m, s_{m-1}, \mathbf{z}_{<m}) || p(x | \mathbf{z}_{\leq m})] - \log p(z_m | \mathbf{z}_{<m}) \right] \\
& = \mathbb{E}_{\mathbf{z}_{<m} | y_m, s_{m-1}} \left[ \frac{1}{\beta_m} D_{\text{KL}} [p(x | y_m, s_{m-1}, \mathbf{z}_{<m}) || p(x | \mathbf{z}_{\leq m})] - \log p(z_m | \mathbf{z}_{<m}) \right]. \tag{F.10}
\end{aligned}$$

## F.4 Calculating required pmfs

This section covers the calculation of the required pmfs for the previously described algorithm. The first term in the KL divergence in (F.10) is calculated by

$$p(x|y_m, s_{m-1}, \mathbf{z}_{<m}) = \frac{p(x, y_m, s_{m-1}, \mathbf{z}_{<m})}{\sum_x p(x, y_m, s_{m-1}, \mathbf{z}_{<m})} \quad (\text{F.11})$$

with

$$p(x, y_m, s_{m-1}, \mathbf{z}_{<m}) = p(z_{m-1}, s_{m-1}|x, \mathbf{z}_{<m-1})p(x, y_m, \mathbf{z}_{<m-1}). \quad (\text{F.12})$$

The first term on the right hand side in (F.12) can be calculated by

$$\begin{aligned} p(z_{m-1}, s_{m-1}|x, \mathbf{z}_{<m-1}) &= \sum_{y_{m-1}} \sum_{s_{m-2}} p(z_{m-1}|y_{m-1}, s_{m-2})p(s_{m-1}|y_{m-1}, s_{m-2})p(y_{m-1}|x) \\ &\quad \cdot p(s_{m-2}|x, \mathbf{z}_{<m-1}) \end{aligned} \quad (\text{F.13})$$

where  $p(z_{m-1}|y_{m-1}, s_{m-2})$  is given as the mapping of previously designed quantizers,  $p(s_{m-1}|y_{m-1}, s_{m-2})$  is given as the predefined mapping of instantaneous side-information (see Appendix E) and  $p(y_{m-1}|x)$  is given as the measurement channel for the previous sensor. The term  $p(s_{m-2}|x, \mathbf{z}_{<m-1})$  can already be calculated by previous sensors (in this case, the pre-predecessor) in a recursive way

$$p(s_m|x, \mathbf{z}_{<m}) = \sum_{y_m} \sum_{s_{m-1}} p(s_m|y_m, s_{m-1})p(y_m|x, z_m)p(s_{m-1}|x, \mathbf{z}_{<m-1}) \quad (\text{F.14})$$

where

$$p(y_m|x, z_m) = \frac{p(y_m, x, z_m)}{\sum_{y_m} p(y_m, x, z_m)} \quad (\text{F.15})$$

with

$$p(y_m, x, z_m) = \sum_{s_{m-1}} p(z_m|y_m, s_{m-1})p(y_m|x)p(s_{m-1}|x)p(x). \quad (\text{F.16})$$

The second term on the right hand side in (F.12) can be calculated by

$$p(x, y_m, \mathbf{z}_{<m-1}) = p(y_m|x)p(\mathbf{z}_{<m-1}, x). \quad (\text{F.17})$$

The last term in (F.17) can again be already calculated by previous sensors (in this case, the pre-predecessor) by

$$p(\mathbf{z}_{\leq m}, x) = \sum_{y_m} \sum_{s_{m-1}} p(z_m|y_m, s_{m-1})p(z_{m-1}, s_{m-1}|x, \mathbf{z}_{<m-1})p(x, y_m, \mathbf{z}_{<m-1}) \quad (\text{F.18})$$

where  $p(z_{m-1}, s_{m-1}|x, \mathbf{z}_{<m-1})$  is already calculated in (F.13) and  $p(x, y_m, \mathbf{z}_{<m-1})$  is already determined in (F.17). The second term in the KL divergence in (F.10) is computed by

$$p(x|\mathbf{z}_{\leq m}) = \frac{p(\mathbf{z}_{\leq m}, x)}{\sum_x p(\mathbf{z}_{\leq m}, x)} \quad (\text{F.19})$$

where  $p(\mathbf{z}_{\leq m}, x)$  is already calculated in (F.18). The term in the logarithm in (F.10) can be determined as

$$p(z_m|\mathbf{z}_{<m}) = \frac{\sum_x p(\mathbf{z}_{\leq m}, x)}{\sum_x \sum_{z_m} p(\mathbf{z}_{\leq m}, x)}. \quad (\text{F.20})$$

Finally, the required pmf to calculate the conditional expectation in (F.10) can be determined by

$$p(\mathbf{z}_{<m}|y_m, s_{m-1}) = \frac{\sum_x p(x, y_m, s_{m-1}, \mathbf{z}_{<m})}{\sum_x \sum_{\mathbf{z}_{<m}} p(x, y_m, s_{m-1}, \mathbf{z}_{<m})} \quad (\text{F.21})$$

in which  $p(x, y_m, s_{m-1}, \mathbf{z}_{<m})$  has already been calculated in (F.12).

Note that all the above equations simplify when optimizing the first sensor to the scalar IB equations given in Section 3.4. Moreover, when optimizing the second sensor, there is no pre-predecessor  $m - 2$  and its impact on the above equations can be omitted.