QUANTISATION EFFECTS IN DISTRIBUTED OPTIMISATION

Joseph A.G. Jonkman, Thomas Sherson and Richard Heusdens

Circuits and Systems group, Delft University of Technology, Delft, the Netherlands

ABSTRACT

In this paper the effects of quantisation on distributed convex optimisation algorithms are explored via the lens of monotone operator theory. Specifically, by representing transmission quantisation via an additive noise model, we demonstrate how quantisation can be viewed as an instance of an inexact Krasnosel'skiĭ-Mann scheme. In the case of two distributed solvers, the Alternating Direction Method of Multipliers and the Primal Dual Method of Multipliers, we further demonstrate how an adaptive quantisation scheme can be constructed to reduce transmission costs between nodes. Finally for the Gaussian channel capacity maximisation problem, we demonstrate convergence even in the presence of one-bit uniform quantisation based on the aforementioned adaptive quantisation scheme.

Index Terms— Quantisation, Monotone Operator Theory, Distributed Convex Optimisation, PDMM, ADMM.

1. INTRODUCTION

In the last two decades, the fields of computer science and signal processing have seen a dramatic increase in the interest in and deployment of distributed computational methods. In areas such as machine learning [1, 2, 3], big data processing [4, 5] and network based signal processing [6, 7, 8] such as 'Internet of Things' type applications, distributed processing has become a ubiquitous approach to parallelising computational efforts across multiple nodes. In particular, by allowing networks of computers to actively work together to solve a common problem, these approaches also inherit numerous advantages in contrast to their centralised counterparts including robustness to node failure and topology changes, scalability with network size, a reduction in power usage and more [9].

Within the literature a number of different approaches for distributed signal processing have been proposed, like distributed averaging [6], gossip [10], message passing based algorithms [11, 12, 13], ADMM and PDMM. A common feature of all of the above mentioned methods, and the one which facilitates distributed computation, is the ability for nodes to communicate with each other and thus exchange information. Whilst in theory these messages are of infinite precision, in practice such signals must be coded and thus quantised. This highlights an important practical artefact which is not often considered during algorithmic derivation, that of a lack of precision of the iterates. Whilst the effects of quantisation on some distributed solvers have been addressed within the literature [14, 15, 16, 17], with a specific effort into the effects on PDMM given in [18], the purpose of this paper is to provide a general model for quantisation in distributed optimisation by exploiting its inherent link with monotone operator theory. Furthermore, by relating the effect of quantisation with an inexact Krasnosel'skiĭ-Mann iterative scheme [19] we demonstrate how adaptive fixed rate quantisers can be developed for use in practical contexts. The effects of such quantisation schemes are also highlighted by demonstrating their use in solving a Gaussian channel capacity maximisation problem with specific attention paid to how the performance of both PDMM and ADMM are effected.

2. BACKGROUND ON MONOTONE OPERATOR THEORY

Many distributed optimisation solvers including the likes of gradient descent, proximal gradient methods, ADMM and PDMM can be derived through the guise of monotone operator theory. This framework therefore provides the ideal foundation for a general analysis of the effect that quantisation has on such solvers. For this purpose, we begin with an initial overview of some of the basis definitions and properties of monotone operators and operator splitting techniques which will be used throughout the text. For a more detailed and self contained overview of this subject the reader is referred to the tutorial paper provided in [20].

2.1. Monotone Operator Theory

An operator T on $\mathbb{R}^n$ is a subset of $\mathbb{R}^n \times \mathbb{R}^n$. We will write $T(x)$ to denote the *image* or *range* of T defined as $T(x) = \{y \in \mathbb{R}^n \,|\, \exists x \in \mathbb{R}^n : (x, y) \in T\}$. If $T(x)$ is a singleton or empty for any x, then T is called a function or single-valued. The inverse relation of T is defined as $T^{-1} = \{(x, y) \,|\, (y, x) \in T\}$. If T_1 and T_2 are any operators, we let $T_1 + T_2 = \{(x, y + z) \,|\, (x, y) \in T_1, (x, z) \in T_2\}$ and $T_2 \circ T_1 = \{(x, z) \,|\, \exists\, y \in \mathbb{R}^n : (x, y) \in T_1, (y, z) \in T_2\}$. For any $\rho \in \mathbb{R}_+$ and operator T, we let ρT be the operator $\{(x, \rho y) \,|\, (x, y) \in T\}$. The *resolvent* of an operator T is defined as

$$J_{\rho T} = (I + \rho T)^{-1}, \tag{1}$$

where I is the identity relation $I = \{(x, x) \,|\, x \in \mathbb{R}^n\}$. The *Cayley operator*, reflection operator, or reflected resolvent of T, is defined as $C_{\rho T} = 2J_{\rho T} - I$.

A relation is called *monotone* if

$$(T(y) - T(x))^T (y - x) \geq 0,$$

and *strongly monotone* or *coercive* with parameter $m > 0$ if

$$(T(y) - T(x))^T (y - x) \geq m\|y - x\|_2^2,$$

for all $x, y \in \mathbb{R}^n$. A monotone operator is called *maximal* if there is no monotone operator that properly contains it. That is, $\nexists\,(x, u) \notin T$ such that $T \cup \{(x, u)\}$ is still monotone. A relation is called *Lipschitz continuous* if there exists a $L \geq 0$ such that

$$\|T(y) - T(x)\|_2 \leq L\|y - x\|_2,$$

for all $x, y \in \mathbb{R}^n$. T is called *non-expansive* if $L \leq 1$ and a *contraction* if $L < 1$. By inspection of Eq. (1), we conclude that if T is monotone, then $I + \rho T$ is strongly monotone so that $J_{\rho T} = (I + \rho T)^{-1}$ is single valued. In fact, $J_{\rho T}$ is Lipschitz continuous with parameter $L \leq 1$ and is therefore non-expansive.

The *fixed point set* of the operator T is given by $\text{fix}(T) = \{x \in \mathbb{R}^n \,|\, x = T(x)\}$. It can be empty, or contain many points. Such a fixed point can be iteratively found by solving

$$x^{(k+1)} = T\big(x^{(k)}\big), \tag{2}$$

where the superscript $^{(\cdot)}$ denotes the iteration number and $x^{(0)} \in \mathbb{R}^n$ is some initialisation. It can be shown that if T is a contraction, the fixed-point algorithm converges to a unique fixed point. In the case that T is non-expansive, by the Krasnosel'skiĭ-Mann theorem [21, Proposition 5.16], the fixed point iterations of the averaged operator $G = (1-\theta)I + \theta T$ for some $\theta \in (0,1)$ will converge to a solution if one exists.

For maximal monotone operators there is an important relation between the zeros, or roots, of said operator and the fixed points of its associated resolvent operator. Notably, we have

$$0 \in T(x) \;\;\Leftrightarrow\;\; x \in (I + \rho T)(x) \;\;\Leftrightarrow\;\; x = J_{\rho T}(x), \tag{3}$$

where the last relation holds since $J_{\rho T}$ is single valued. Moreover, if $x \in \text{fix}(J_{\rho T})$, then $x \in \text{fix}(C_{\rho T})$.

2.2. Convex Optimisation and Monotone Operator Theory

As previously alluded to, monotone operator theory can be used to solve convex optimisation problems. For instance, let the function $f : \mathbb{R}^n \mapsto \mathbb{R} \cup \{\infty\}$ be closed, convex and proper (CCP). Consider solving the unconstrained optimisation problem given by

$$\min_x \; f(x). \tag{4}$$

The vector x^* is a minimiser of Eq. (4) if and only if $0 \in \partial f(x^*)$, where ∂f is the *subdifferential* relation of f defined by

$$\partial f = \{(x, g) \,|\, x \in \mathbb{R}^n,\; \forall y \in \mathbb{R}^n \,:\, f(y) \geq f(x) + g^T(y - x)\},$$

which is monotone [20]. It follows from Eq. (3) that instead of finding a zero of ∂f, we can find a fixed point of its associated resolvent. When combined with Eq. (2), this allows us to solve convex optimisation problems via an iterative approach.

2.3. Operator Splitting

The main difficulty in fixed-point methods is the evaluation of the resolvent. In particular the inversion operation needed to evaluate the resolvent may be prohibitively expensive to compute. One way to overcome this problem is to decompose the operator T into two maximal monotone operators T_1 and T_2 such that $T = T_1 + T_2$, where the resolvents of T_1 and T_2 are easier to evaluate. Examples of operator splitting include Peaceman-Rachford splitting and Douglas-Rachford splitting, with the latter providing the basis for ADMM which can be seen as applying Douglas-Rachford splitting to the dual function [20]. Peaceman-Rachford splitting is given by

$$0 \in T_1(x) + T_2(x) \;\Leftrightarrow\; z = C_{\rho T_2} \circ C_{\rho T_1}(z),\, x = J_{\rho T_1}(z),$$

where the introduced z variables will be referred to as *auxiliary variables* from here on out. As the Cayley operator is non-expansive, it follows that the operator $C_{\rho T_2} \circ C_{\rho T_1}$ is non-expansive and induces the following fixed point iteration

$$z^{(k+1)} = C_{\rho T_2} \circ C_{\rho T_1}(z^{(k)}).$$

Since $C_{\rho T_2} \circ C_{\rho T_1}$ is non-expansive, there is no guarantee that the iterates will converge. To ensure convergence without imposing additional conditions like strong monotonicity and Lipschitz continuity, we can average the non-expansive operator such that

$$z^{(k+1)} = \big((1-\alpha)I + \alpha C_{\rho T_2} \circ C_{\rho T_1}\big)(z^{(k)}), \tag{5}$$

where $\alpha \in (0,1)$. The effect of such averaging guarantees convergence to an optimal solution, if one exists [21]. For $\alpha = \frac{1}{2}$, the algorithm is referred to as Douglas-Rachford splitting.

3. INCORPORATING QUANTISATION INTO THE MONOTONE OPERATOR FRAMEWORK

An inherent characteristic of distributed solvers is the need for nodes to transmit and share data within the network. In practical systems this results in the inevitable introduction of quantisation noise due to the use of finite precision transmission schemes. In this way, it is interesting to consider the impacts that such quantisation may have on an algorithms performance. Alternatively one can deliberately introduce further quantisation to facilitate a reduction in transmission costs between nodes by reducing the size of transmitted packets. To explore the impact that quantisation can have, we directly model quantisation noise via the use of an inexact Krasnosel'skiĭ-Mann iterative scheme[19].

3.1. Inexact Krasnosel'skiĭ-Mann iterations

The inexact Krasnosel'skiĭ-Mann iterations are given by

$$\begin{aligned} z^{(k+1)} &= z^{(k)} + \alpha^{(k)}\big(T\big(z^{(k)}\big) + \epsilon^{(k)} - z^{(k)}\big) \\ &= \big((1-\alpha^{(k)})I + \alpha^{(k)}T\big)\big(z^{(k)}\big) + \alpha^{(k)}\epsilon^{(k)}, \end{aligned} \tag{6}$$

which has clear similarities to Eq. (5). In fact, if $\alpha^{(k)} = \frac{1}{2}\;\; \forall k$ and $T = C_{\rho T_2} \circ C_{\rho T_1}$, Eq. (6) reduces to

$$z^{(k+1)} = \frac{1}{2}\big(I + C_{\rho T_2} \circ C_{\rho T_1}\big)\big(z^{(k)}\big) + \frac{1}{2}\epsilon^{(k)}.$$

Hence, inexact Douglas-Rachford splitting is equivalent to an inexact Krasnosel'skiĭ-Mann scheme, where $\frac{1}{2}\epsilon^{(k)}$ describes an additive error term, for example due to quantisation. Peaceman-Rachford splitting can similarly be cast as inexact Krasnosel'skiĭ-Mann iterations in the case when $\alpha^{(k)} = 1\;\; \forall k$. Under the assumption that the sequence of errors are finitely summable, i.e. that $(e^{(k)})_{k\in\mathbb{N}} \in \ell^1_+$, and that the averaging terms satisfy $\big((\alpha^{(k)}(1-\alpha^{(k)})\big)_{k\in\mathbb{N}} \notin \ell^1_+$ it was demonstrated in [19, Proposition 1(iii)] that the sequence $(z^{(k)})_{k\in\mathbb{N}}$ converges to some $z^* \in \text{fix}\,(T)$.

An immediate consequence of this result is that whilst the rate of convergence of a distributed solver may be influenced by the presence of quantisation, it is possible for such algorithms to still converge. In the following we highlight a method for designing appropriate quantisers for distributed optimisation based on this fact.

3.2. Quantisation in Distributed Optimisation

As previously mentioned, monotone operator theory and operator splitting form the basis of many distributed optimisation algorithms. Specifically, in the later portion of this paper we will show that PDMM and ADMM can be expressed via Peaceman-Rachford and Douglas-Rachford splitting respectively. In the following T' will be used to denote either of the Peaceman-Rachford or Douglas-Rachford operators. For such operators, assume that a scalar quantiser is used to encode the iterates such that the quantised update equations are given by

$$z^{(k+1)} = T'\big(z^{(k)}\big) + n_q^{(k)} = \tilde{z}^{(k+1)} + n_q^{(k)},$$

where the vector $n_q^{(k)}$ denotes the quantisation noise introduced in the k-th iteration. In particular, the vector $n_q^{(k)}$ is dependent on the vector $\tilde{z}^{(k+1)} = T'(z^{(k)})$. As the operator T' is non-expansive, this leads to the fact that

$$\|z^{(k+1)} - z^*\| \leq \|T'\big(z^{(k)}\big) - z^*\| + \|n_q^{(k)}\|.$$

As mentioned before, if the sequence $\big(\|n_q^{(k)}\|\big)_{k\in\mathbb{N}}$ is finitely summable $\|z^{(k+1)} - z^*\|$ must converge. Furthermore, in the case that T' is the Douglas-Rachford operator, it was shown in [21] that $z^{(k)}$ must converge to a fixed point of T'. Without further consideration, this indicates that a quantiser of increasing precision will be required to guarantee convergence. Practically speaking this is an undesirable feature as it corresponds to an increasing transmission rate as the number of iterations increases.

3.3. Redefining the Transmissions

To avoid the need for such a dynamic rate quantiser, we highlight a simple observation motivated by the work in [18]. For algorithms which converge to a fixed point, the entropy of the updates decreases with increasing iterations. This suggests that the rate of messages should be able to be decreased with increasing iteration count, a direct contradiction of the aforementioned requirement for guaranteed convergence. In this work we consider the use of a predefined sequence of quantisers to allow for low data rate transmission between nodes to occur whilst maintaining the convergent nature of the algorithm.

A fundamental limitation of such a sequential quantisation technique is that without knowing the fixed point in question to which the algorithm converges in advance, such quantisers cannot be practically deployed. To address this point, we propose to instead quantise and transmit the difference of the iterates which converges to the zero vector. This difference vector is given by $v^{(k+1)} = \tilde{z}^{(k+1)} - z^{(k)}$ such that $z^{(k+1)} = z^{(k)} + v^{(k+1)}$. To define an appropriate quantiser therefore only requires the knowledge of the distribution of $v^{(k)}$. Whilst this still cannot be known exactly a priori, in certain problems we can infer a bounding distribution on this variable allowing us to design a sequence of quantisers prior to algorithm deployment.

To demonstrate this point, consider the covariance matrix of the transmitted variables (in this case $v^{(k+1)}$) which is of importance as it directly determines the entropy of each message. In the case of a scalar uniform quantiser, an upper bound on this entropy is given by

$$H(X) \leq \frac{1}{2}\log\left(\frac{2\pi e\sigma_X^2}{\Delta^2}\right),$$

where Δ denotes the cell width or precision of the quantiser and σ_X^2 denotes the variance of X. In the case of quantising $v^{(k+1)}$, this covariance can be upper bounded by

$$\operatorname{cov}\big(v^{(k)}\big) = \mathbb{E}\Big[v^{(k)}{v^{(k)}}^H\Big] - \mathbb{E}\Big[v^{(k)}\Big]\mathbb{E}\Big[{v^{(k)}}^H\Big] \leq \mathbb{E}\Big[\|v^{(k)}\|^2\Big]I.$$

Importantly in inexact Krasnosel'skiĭ-Mann iterations $\|v^{(k)}\|^2$ forms a monotonically decreasing sequence. Thus, if the worst case rate of decrease of $\|v^{(k)}\|^2$ is known, the same decrease in cell width can be implemented to maintain a fixed bit rate for the quantisation, whilst simultaneously ensuring that the sequence $\big(\|n_{q,z^{(k)}}\|\big)_{k\in\mathbb{N}}$ is finitely summable. In the case of ADMM or Douglas-Rachford splitting, this in turn guarantees the convergence of the algorithm to a true fixed point [20]. Whilst the same is not necessarily true for PDMM or Peaceman-Rachford splitting, in practice it is observed that the same result holds.

4. APPLICATION

To demonstrate the performance of this approach, in the following we apply a sequentially quantised approach to the task of *Gaussian channel capacity* (GCC) maximisation. Such a problem is often found in multi-channel information theory analysis and can be directly solved in a centralised context via the well known waterfilling method [22].

As demonstrated in [23], the GCC problem for N channels can be cast as an equivalent convex optimisation problem given by

$$\begin{aligned} \min_{x} \quad & -\sum_{i=1}^{N}\log(\sigma_i^2 + x_i) \\ \text{s.t.} \quad & x \geq 0,\ \sum_{i=1}^{N} x_i = p_{\text{tot}}, \end{aligned} \tag{7}$$

where $\sigma_i^2 > 0, x_i \geq 0\ \forall\ i = 1, ..., N$ denote the noise and allocated power for each channel respectively and the scalar p_{tot} denotes the total power requirement of the system. In a transmit beamforming type application, each channel may represent a node within a network. The connectivity within such a network can be represented by its graphical form $G(V, E)$ where V denotes the set of vertices (nodes) and E denotes the set of edges (inter-node communication paths) such that $(i, j) \in E$ only if nodes i and j can communicate. In particular, as the local noise powers σ_i may be estimated locally at each node, the objective is to devise a way to solve this problem in a distributed manner to remove the need for a data aggregation step.

In [24] it was shown that while Eq. (7) is not immediately distributable in the primal domain, it is distributable in the dual domain. A distributed variant of the Lagrange dual of Eq. (7) is given by

$$\begin{aligned} \min_{\lambda,\mu} \quad & \sum_{i\in V}\left(\log\Big(\frac{-1}{\mu_i + \lambda_i}\Big) - \mu_i\sigma_i^2 - \lambda_i\sigma_i^2 - \frac{\mu_i}{N}\right) \\ \text{s.t.} \quad & \mu_i = \mu_j,\ \ \forall (i,j) \in E \\ & 0 > \mu_i + \lambda_i \forall i \in V \\ & 0 \leq \lambda_i\ \ \forall i \in V. \end{aligned} \tag{8}$$

where the introduced λ and μ variables are the dual variables for the inequality and equality constraints in Eq. (7) respectively.

We will now consider the task of solving this problem via two different algorithms, PDMM and ADMM whilst incorporating the proposed sequential quantisation approach outlined previously. In particular, as the observed convergence rate of both algorithms is geometric, in this case a geometrically contracting cell width $\Delta^{(k+1)} = \gamma^k\Delta^{(0)}$ was used for some initial cell width $\Delta^{(0)}$ and $\gamma \in [0, 1)$.

4.1. Primal-Dual Method of Multipliers

As previously mentioned, the PDMM algorithm proposed in [25] can be derived from a monotone operator perspective. In particular, it is equivalent to Peaceman-Rachford splitting applied to the Lagrange dual of a distributed problem. As such, for the particular GCC problem considered the PDMM updates are given by

$$(\lambda_i^{(k+1)}, \mu_i^{(k+1)}) = \arg\min_{\lambda_i,\mu_i} (q_i(\lambda_i, \mu_i, z^{(k)}))$$
$$\text{s.t. } \lambda_i \geq 0,\ \ \mu_i + \lambda_i < 0\ \ \forall i \in V \quad (9a)$$
$$\beta_{i|j}^{(k+1)} = z_{i|j}^{(k)} + \rho \frac{i-j}{|i-j|}\mu_i^{(k+1)} \quad (9b)$$
$$y_{i|j}^{(k+1)} = 2\beta_{i|j}^{(k+1)} - z_{i|j}^{(k)} \quad (9c)$$
$$z_{i|j}^{(k+1)} = y_{j|i}^{(k+1)}, \quad (9d)$$

where

$$q_i(\lambda_i, \mu_i, z^{(k)}) = \log\left(\frac{-1}{\mu_i + \lambda_i}\right) - \mu_i\sigma_i^2 - \lambda_i\sigma_i^2 - \frac{\mu_i p_{\text{tot}}}{N} + \sum_{j\in\mathcal{N}(i)} \left(\frac{i-j}{|i-j|} z_{i|j}^{(k)^T}\mu_i + \frac{\rho}{2}\|\mu_i\|_2^2\right).$$

and $\mathcal{N}(i) = \{j \mid (i,j) \in E\}$. Importantly, Eq. (9d) can be interpreted as a transmission of data between nodes. This is indicated by the change in indexing where the notation $i|j$ is used to denote a variable related to the directed edge from node i to node j. For instance the transmission made from not i to node j would be given by $v_{j|i}^{(k+1)} = y_{i|j}^{(k+1)} - z_{j|i}^{(k)}$. It is at this stage that quantisation noise will be introduced and our sequential quantisation technique exploited. It should be noted that this difference term requires the knowledge of $z_{j|i}^{(k)}\ \forall\, j \in \mathcal{N}(i)$. Fortunately, $\forall i$ this value can be locally tracked at each node by updating a local reference variable $\hat{z}^{(k)}$ with the same quantised $v_{j|i}^{(k+1)}$ transmissions.

4.2. Alternating Direction Method of Multipliers

While PDMM can be shown to be equivalent to Peaceman-Rachford splitting, as previously mentioned, ADMM is equivalent to Douglas-Rachford splitting applied to the dual problem. Thus, the ADMM updates are equal to that of PDMM, except that the z update given in Eq. (9d) becomes

$$z_{i|j}^{(k+1)} = \frac{1}{2}\left(z_{i|j}^{(k)} + y_{j|i}^{(k+1)}\right).$$

Furthermore, the transmissions made between nodes as part of the ADMM algorithm are identical to those used in PDMM and require the same local tracking of the neighbouring $z_{j|i}^{(k)}$ variables.

5. SIMULATIONS AND RESULTS

The above quantised distributed methods were both used to solve the distributed dual GCC problem given in Eq. (8). For this purpose, a random geometric graph [26] of 50 nodes was constructed where the transmission radius of each node was set to $\sqrt{\frac{\log(N)}{N}}$ to ensure a connected graph with high probability [26]. The resulting network was verified as forming a single connected component to prevent partitioning of the network. The available total transmission power was set such that $p_{\text{tot}} = 16$ and the set of $\sigma_i^2,\ i \in V$ were drawn from a uniform distribution between 0 and 0.4.

In the considered instance, the penalty parameter ρ was selected per algorithm to optimise convergence rate in the non-quantized case whilst the cell width contraction factor γ was chosen to optimise the quantised convergence rate using the same ρ. Furthermore, the initial cell width was set such that $\Delta^{(0)} = 1$ and a fixed one-bit quantiser

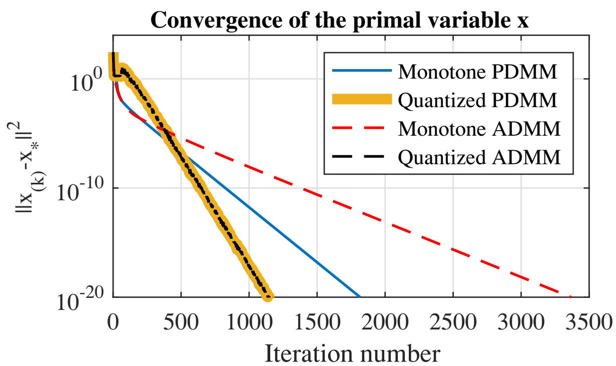

Fig. 1. Convergence of the primal variable of the GCC problem for different algorithms. For the quantised algorithms, a one-bit quantiser was used .

was used for all simulations. Fig. 1 compares the convergence rate of the primal variable $x^{(k)}$ for both algorithms in both the cases of non-quantized distributed optimisation and the equivalent optimisation approach.

As expected both the quantised and non-quantized variants of both algorithms converge to the true x^* as computed by the standard water filling approach. In the non-quantized case, we can note that with optimal ρ selection PDMM converges faster than ADMM, requiring around half the number of iterations to achieve the same primal precision. This speedup is expected to be related to the lack of averaging used in the case of PDMM. However, once sequential quantisation is applied to both algorithms, convergence occurs at almost the same rate for both ADMM and PDMM. The rate of convergence, when the quantisation is applied, is in fact faster than that of the non-quantized case, achieving the same precision in around 30% fewer iterations. This raises a somewhat counter-intuitive point, that reducing the information rate between nodes can increase the rate of algorithmic convergence if correctly implemented. A potential justification for this is that whilst information is discarded during the quantisation process, the essence of the update direction is preserved. Thus redundant information may be discarded, in turn aiding convergence. A more mathematical exploration of this point is beyond the scope of this paper and is left for future work.

6. CONCLUSION

In this paper the effects of quantisation on distributed convex optimisation algorithms were explored via the lens of monotone operator theory. By modelling the effects of quantisation noise as an implementation of an inexact Krasnosel'skiĭ-Mann scheme sufficient conditions for convergence were established. Furthermore, by exploiting the observation that the entropy of the iterates of such distributed methods decreases with iteration count a sequential quantisation mechanism was proposed to achieve constant rate transmission and high final precision. Such a theory was demonstrated in the context of the Gaussian channel capacity maximisation problem applied to two distributed solvers, the Alternating Direction Method of Multipliers and the Primal Dual Method of Multipliers. Interestingly it was shown that with appropriate tuning, the convergence rate of both algorithms could not only be maintained but also accelerated, leading to a reduction in the number of iterations required to reach a desired precision. Overall this motivates the point that quantisation for distributed algorithms is not only beneficial from a transmission cost perspective but also may be able to be used to improve algorithmic performance.

7. REFERENCES

[1] David E Goldberg and John H Holland, "Genetic algorithms and machine learning," *Machine learning*, vol. 3, no. 2, pp. 95–99, 1988.

[2] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends® in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.

[3] Ron Bekkerman, Mikhail Bilenko, and John Langford, *Scaling up machine learning: Parallel and distributed approaches*, Cambridge University Press, 2011.

[4] Changqing Ji, Yu Li, Wenming Qiu, Uchechukwu Awada, and Keqiu Li, "Big data processing in cloud computing environments," in *Pervasive Systems, Algorithms and Networks (ISPAN), 2012 12th International Symposium on*. IEEE, 2012, pp. 17–23.

[5] Volkan Cevher, Stephen Becker, and Mark Schmidt, "Convex optimization for big data: Scalable, randomized, and parallel algorithms for big data analytics," *IEEE Signal Processing Magazine*, vol. 31, no. 5, pp. 32–43, 2014.

[6] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future generation computer systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[7] A. Simonetto and G. Leus, "Distributed maximum likelihood sensor network localization," *IEEE Trans. Signal Processing*, vol. 62, no. 6, pp. 1424–1437, 2014.

[8] J. Mota, J. Xavier, P. Aguiar, and M. Puschel, "Distributed ADMM for model predictive control and congestion control," in *IEEE Conf. Decision and Control (CDC)*. IEEE, 2012, pp. 5110–5115.

[9] Michael G Rabbat and Robert D Nowak, "Quantized incremental algorithms for distributed optimization," *IEEE Journal on Selected Areas in Communications*, vol. 23, no. 4, pp. 798–808, 2005.

[10] Stephen Boyd, Arpita Ghosh, Balaji Prabhakar, and Devavrat Shah, "Randomized gossip algorithms," *IEEE/ACM Transactions on Networking (TON)*, vol. 14, no. SI, pp. 2508–2530, 2006.

[11] Judea Pearl, "Fusion, propagation, and structuring in belief networks," *Artificial intelligence*, vol. 29, no. 3, pp. 241–288, 1986.

[12] Martin J Wainwright, Michael I Jordan, et al., "Graphical models, exponential families, and variational inference," *Foundations and Trends® in Machine Learning*, vol. 1, no. 1–2, pp. 1–305, 2008.

[13] Ciamac C Moallemi and Benjamin Van Roy, "Convergence of min-sum message passing for quadratic optimization," *IEEE Transactions on Information Theory*, vol. 55, no. 5, pp. 2413–2423, 2009.

[14] Tuncer Can Aysal, Mark J Coates, and Michael G Rabbat, "Distributed average consensus with dithered quantization," *IEEE Transactions on Signal Processing*, vol. 56, no. 10, pp. 4905–4918, 2008.

[15] Tao Li, Minyue Fu, Lihua Xie, and Ji-Feng Zhang, "Distributed consensus with limited communication data rate," *IEEE Transactions on Automatic Control*, vol. 56, no. 2, pp. 279–292, 2011.

[16] Soummya Kar and José MF Moura, "Distributed consensus algorithms in sensor networks: Quantized data and random link failures," *IEEE Transactions on Signal Processing*, vol. 58, no. 3, pp. 1383–1400, 2010.

[17] Angelia Nedic, Alex Olshevsky, Asuman Ozdaglar, and John N Tsitsiklis, "Distributed subgradient methods and quantization effects," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*. IEEE, 2008, pp. 4177–4184.

[18] Daan HM Schellekens, Thomas Sherson, and Richard Heusdens, "Quantisation effects in pdmm: A first study for synchronous distributed averaging," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 2017, pp. 4237–4241.

[19] Jingwei Liang, Jalal Fadili, and Gabriel Peyré, "Convergence rates with inexact non-expansive operators," *Mathematical Programming*, vol. 159, no. 1-2, pp. 403–434, 2016.

[20] Ernest K Ryu and Stephen Boyd, "Primer on monotone operator methods," *Appl. Comput. Math*, vol. 15, no. 1, pp. 3–43, 2016.

[21] Heinz H Bauschke and Patrick L Combettes, *Convex analysis and monotone operator theory in Hilbert spaces*, Springer Science & Business Media, 2011.

[22] Kenneth W Shum, Kin-Kwong Leung, and Chi Wan Sung, "Convergence of iterative waterfilling algorithm for gaussian interference channels," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 6, pp. 1091–1100, 2007.

[23] Stephen Boyd and Lieven Vandenberghe, *Convex optimization*, Cambridge university press, 2004.

[24] Thomas Sherson, Richard Heusdens, and W Bastiaan Kleijn, "On the duality of globally constrained separable problems and its application to distributed signal processing," in *Signal Processing Conference (EUSIPCO), 2016 24th European*. IEEE, 2016, pp. 1083–1087.

[25] Guoqiang Zhang and Richard Heusdens, "Distributed optimization using the primal-dual method of multipliers," *IEEE Trans. on Signal and Information Processing over Networks*, 2016, Accepted for publication.

[26] Jesper Dall and Michael Christensen, "Random geometric graphs," *Physical Review E*, vol. 66, no. 1, pp. 016121, 2002.