Large-Scale Multi-Processor Approximate Message Passing with Lossy Compression

Dror Baron*
Joint w/ Junan Zhu,* Puxiao Han,# Ruixin Niu,# and Ahmad Beirami+

*North Carolina State University
#Virginia Commonwealth University
+MIT and Duke University

Information Theory and Applications Workshop
San Diego, CA; Jan.31-Feb.5, 2016

Supported by NSF and ARO
Motivation
• Input signal $\mathbf{x} \in \mathbb{R}^N$
• Measurements taken by matrix $\mathbf{A} \in \mathbb{R}^{M \times N}$
• Additive channel noise $\mathbf{z}$
• Measurement rate $\kappa = \frac{M}{N} > 0$
Linear Inverse Problem

\[ y = A x + z \]

- Need to estimate \( x \) from \( y \), \( A \), and statistical info on \( x \), \( z \)
Applications of Linear Inverse Problems

- Medical Imaging
- CDMA
- Seismic Imaging
- Financial Prediction
Approximate Message Passing (AMP)
Approximate Message Passing (AMP)  
[Donoho et al. 2009]

\[ y = Ax + z \]

Residual
\[ r^t = y - Ax^t + \frac{r^{t-1}}{\kappa} < d\eta_t(x^{t-1} + A^T r^{t-1}) > \]

Pseudo-data
\[ f^t = x^t + A^T r^t = x + w^t \]

Denoising
\[ x^{t+1} = \eta_t(f^t) \]

State evolution
\[ \sigma_{t+1}^2 = \sigma_Z^2 + \frac{1}{\kappa} \text{MSE}(\eta_t, \sigma_t^2) \]
Multi-Processor AMP (MP-AMP)
Multi-Processor Linear System
[Patterson et al. 2013, Han et al. 2014]

Matrix $\mathbf{A}$ could be big
$\mathbf{A}$ stored in distributed nodes
Node $p$ processes $y_p = z_p + \mathbf{A}_p \mathbf{x}$
Multi-Processor AMP (MP-AMP)  
[Han et al. 2014]

Centralized AMP
- Calculate residual $r_t$
- Calculate pseudo-data $f_t$
- Denoise $f_t$

MP-AMP
- $P$ distribute nodes
- $r^p_t$: residual in node $p$
- $f^p_t$: pseudo-data in node $p$

$f_t = \sum_{p=1}^{P} f^p_t$

Denoise $f_t$

Node 1

Node $P$

Fusion center
MP-AMP

- Messages: uplink $f_t^p$ and downlink $x_{t+1}$
- Compress messages to reduce communication
- Focus on lossy compression of $f_t^p$
- Lossy compression of $x_{t+1}$ - future work
Lossy MP-AMP
Rate-Distortion Theory

[Berger, 1971; Cover & Thomas, 2006]

\[ f_t^p \in \mathbb{R}^N \quad \text{Quantize} \quad Q(f_t^p) \in \mathbb{R}^N \]

- **R**: Rate (bits/entry) to encode \( Q(f_t^p) \)
- **D**: Distortion between \( f_t^p \) and \( Q(f_t^p) \)
- **R** and **D** related through R-D function
- *Allowing modest D can save lots of R!*
Lossy MP-AMP

\[ Q(f_t^p) = \frac{1}{P}x + w_t^p + n_t^p \] encode w/ \( R \)

State evolution (SE)
\[ \sigma_{t+1}^2 = \sigma_Z^2 + \frac{1}{\kappa} \text{MSE}(\eta_t, \sigma_t^2) \]

Lossy SE
\[ \sigma_{t+1}^2 = \sigma_Z^2 + \frac{1}{\kappa} \text{MSE}(\eta_t, \sigma_t^2 + PD) \]

RD relation \( D = D(R) \)
Question

Lower $R \rightarrow$ higher $D \quad \Rightarrow \quad$ So what?

*Use different $R_t$ in each iteration $t$*

Fix

Aggregate rate $R_{agg}$

# iterations $T \quad \Rightarrow \quad$ What $(R_t)$ minimizes MSE?
Dynamic Programming for Optimal Rates
Dynamic Programming (DP) Scheme
[Bertsekas, 1995]

• Remaining coding rate:

\[ R_{rem}(t) = R_{agg} - \sum_{t=1}^{t-1} R'_{\tilde{t}} \]

• Cost \( \Psi_i(R_{rem}(t), \sigma^2_t) = \text{MSE at output} \)
  s.t. \( i = T - t \) more iterations; \( R_{rem} \) remaining; \( \sigma^2 \) variance
Compute $\Psi_0(R_{rem}(T), \sigma_T^2)$, $\forall R_{rem}(T), \sigma_T^2$
DP Recursion

- Recursion: $\Psi_{T-t}(\text{current}) = \min_{R'} \Psi_{T-(t+1)}(\text{next})$

- Find next with $R'$ and lossy state evolution
- MSE in next has been calculated $\Rightarrow$ computationally efficient
DP Recursion

\[ t = T \]

\[ t = T - 1 \]

\[ t = 1 \]
Numerical Results
Numerical Approach

- Run DP to get optimal coding rate sequences
- Simulate MP-AMP w/optimal coding rates
Approach MMSE with *Reduced* Rate

\[ R_{agg} = 20 \text{ bits}, T = 10, P = 100, x_i \sim 0.1N(0,1) + 0.9\delta(x_i), N = 20,000 \]

- Single precision: 32 bits \( \times T = 320 \) bits
- \( R_{agg} = 20 \) bits can often approach MMSE
Approach MMSE with *Modest* Rate

- Greater rates sometimes needed

\[
\begin{align*}
R_{\text{agg}} &= 222, \\
T &= 122, \\
\mathcal{P} &= 12222, \\
\mathcal{x}_i &\sim 1.1 \mathcal{N}, \\
\mathcal{N} &= 222222
\end{align*}
\]

Needs larger \( T \)
Toward Real-World Costs
Real-World Costs

• Real costs?
  – Computation cost
  – Communication cost
  – Other?

• Computation ($C_1$) and communication ($C_2$)
  costs vary

• What is smallest cost to achieve target MSE?

• Design DP with cost $\Psi = C_1T + C_2R_{agg}$
More iterations, less coding rate

Computation costly More coding rate, few # iterations

Communication costly More iterations, less coding rate

Pareto Optimal Curves

[Das & Dennis, 1998]

\[ \Delta = 10 \log \frac{\text{MSE}}{\text{MMSE}} \]
Thank you!
References