A Bayesian Approach for Asynchronous Parallel Sparse Recovery

Alireza Zaeemzadeh⋆, ***Jamie Haddock***†, **Nazanin Rahnavard**⋆, **Deanna Needell**†

† Department of Mathematics, University of California, Los Angeles, {jhaddock, deanna}@math.ucla.edu

⋆ School of Electrical Engineering and Computer Science, University of Central Florida, {zaeemzadeh, nazanin}@eecs.ucf.edu

Sparse Recovery (SR) Problem

Reconstruct s-sparse signal $\boldsymbol{x} \in \mathbb{R}^N$ from few nonadaptive, linear, and noisy measurements, $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{z}$, where $\boldsymbol{A} \in \mathbb{R}^{m\times N}$ is the measurement matrix and $\boldsymbol{z} \in \mathbb{R}^m$ is noise.

Approach

We solve SR in an asynchronous manner, while reducing the effects of *slow* processors on the estimated signal. We solve

$$\min_{\hat{\boldsymbol{x}}\in\mathbb{R}^N} \frac{1}{M}\sum_{B=1}^{M}\frac{1}{2b}\|\boldsymbol{y}_B - \boldsymbol{A}_B\hat{\boldsymbol{x}}\|_2^2 \quad \text{subject to} \quad \|\hat{\boldsymbol{x}}\|_0 \le s,$$

where $\boldsymbol{y}_B$ and $\boldsymbol{A}_B$ are M non-overlapping sub-vectors and sub-matrices of $\boldsymbol{y}$ and $\boldsymbol{A}$. In each iteration, each processor solves one of the sub-problems defined by $\boldsymbol{y}_B$ and $\boldsymbol{A}_B$ and then shares estimated signal information between processors via a Bayesian framework.

Bayesian Framework

Hidden variables ($\mathcal{H}$):

1. *Tally score*, $\phi_n \in [0,1]$, denoting the probability that coefficient n is in support of signal $\boldsymbol{x}$.
2. *Reliability score* for each processor, $r_i \in [0,1]$, denoting the trustworthiness of the measurements of processor i.
3. *Observation reliability*, $u_{ni} \in \{0,1\}$, which indicates if support coefficient n in the estimated signal reported by processor i is reliable.

Observed variables ($\mathcal{D}$):

1. The *support observations*, o_{ni} indicate if processor i detects coefficient n in the support of the estimated signal.
2. The maximum *number of iterations* completed by any processor since the last reporting of processor i, k_i.

Generative model:

$$\begin{aligned}
r_i &\sim \text{Beta}(\beta_i^1, \beta_i^0)\\
u_{ni} &\sim \text{Bernoulli}(r_i)\\
\phi_n &\sim \text{Beta}(a_n^1, a_n^0)\\
o_{ni} &\sim u_{ni}\,\text{Bernoulli}(\phi_n)\\
&\quad + (1-u_{ni})\,\text{Bernoulli}(1-\phi_n)\\
k_i &\sim \text{Binomial}(K_i, r_i)
\end{aligned}$$

We use the generative model to infer the posterior probability distribution of $\mathcal{H}$ from $\mathcal{D}$ using Bayes' rule:

$$\mathbb{P}\{\mathcal{H}|\mathcal{D}\} \propto \mathbb{P}\{\mathcal{D}|\mathcal{H}\}\mathbb{P}\{\mathcal{H}\} = \mathbb{P}\{\mathcal{D},\mathcal{H}\}.$$

where $\mathbb{P}\{\mathcal{D},\mathcal{H}\}$ is calculated using the model above. To avoid intractable computations, we approximate $\mathcal{P}(\mathcal{H})$ by a fully factorized distribution,

$$\mathbb{Q}\{\mathcal{H}\} = \prod_i \mathbb{Q}\{r_i|\hat{\beta}_i^1, \hat{\beta}_i^0\}\prod_{n,i}\mathbb{Q}\{u_{ni}|\tau_{ni}\}\prod_n \mathbb{Q}\{\phi_n|\hat{a}_n^1, \hat{a}_n^0\}.$$

Bayesian Asynchronous StoIHT

Require: Number of subproblems, M, and probability of selection $p(B)$. The parameters of the reliability score, $\hat{\beta}_i^1$ and $\hat{\beta}_i^0$, and the parameters of tally scores, $\hat{a}_n^1$ and $\hat{a}_n^0$, are available to each processor. Each processor performs the following at each iteration:

1: **randomize:** select $B_t \in [M]$ with probability $p(B_t)$

2: **proxy:** $\boldsymbol{b}^{(t)} = \boldsymbol{x}^{(t)} + \frac{\gamma}{Mp(B_t)}\boldsymbol{A}_{B_t}^*(\boldsymbol{y}_{B_t} - \boldsymbol{A}_{B_t}\boldsymbol{x}^{(t)})$

3: **identify:** $\hat{\mathcal{S}}^{(t)} = supp_s(\boldsymbol{b}^{(t)})$ and $\tilde{\mathcal{T}}^{(t)} = supp_s(\phi)$

4: **estimate:** $\boldsymbol{x}^{(t+1)} = \boldsymbol{b}^{(t)}_{\hat{\mathcal{S}}^{(t)}\cup\tilde{\mathcal{T}}^{(t)}}$

5: **repeat**

6: update hidden variable parameters, u_{ni}, $\hat{\beta}_i^1$, $\hat{\beta}_i^0$, $\hat{a}_n^1$, and $\hat{a}_n^0$

7: **until** convergence

8: update ϕ

9: $t = t + 1$

Numerical Experiments

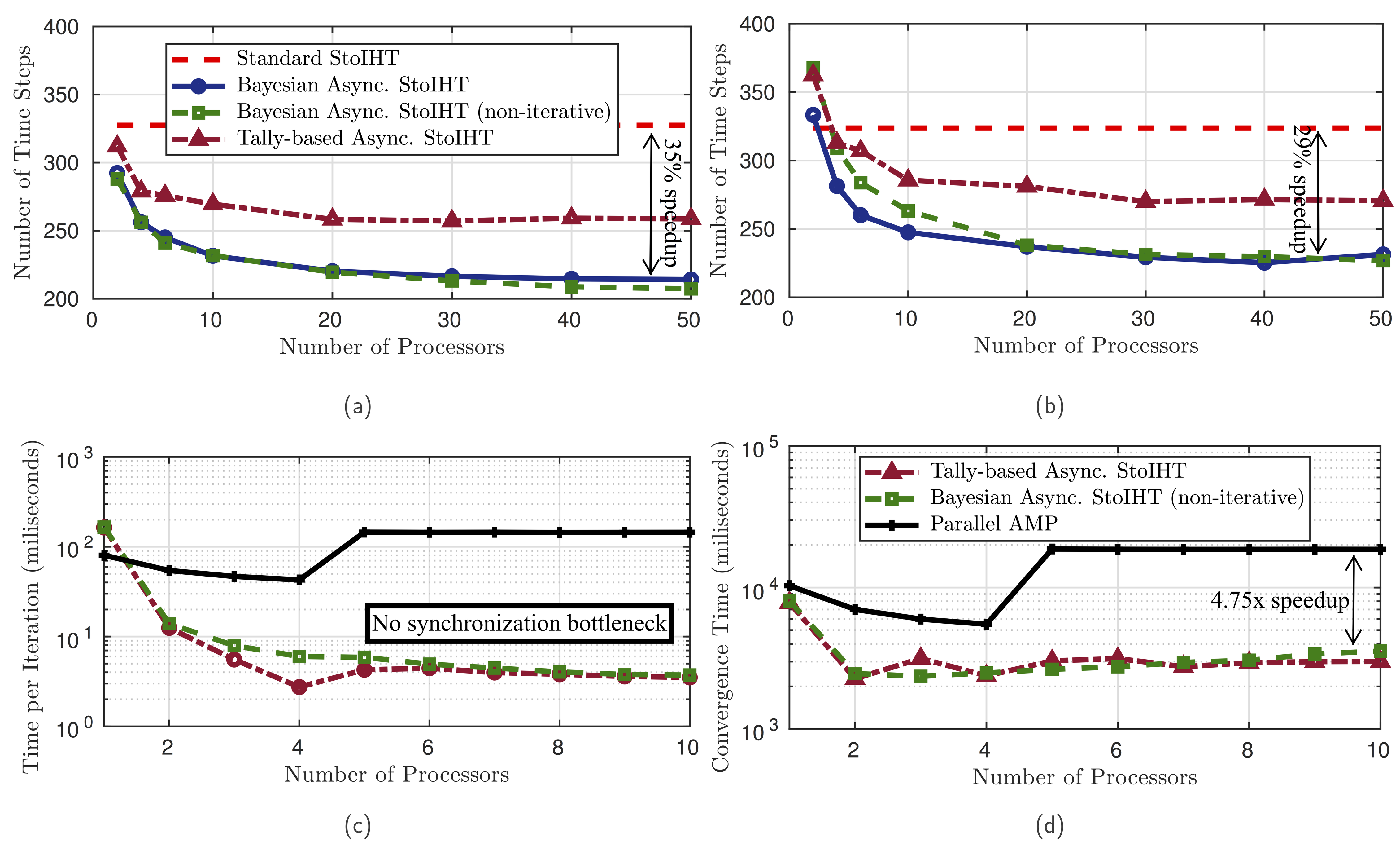

Figure: Comparison of the number of time steps executed until convergence versus the number of processors used in different sparse recovery methods, when (a) all processors are simulated to complete an iteration in a single time step and (b) half of the processors are *slow* and complete an iteration only once out of every four time steps. Performance of different multi-processor sparse recovery algorithms implemented using C++ programming language and OpenMP platform measured in (c) time per iteration and (d) convergence time. Here 20% of the processors are slow.

Conclusions

We proposed an asynchronous stochastic thresholding approach to solving the SR problem which reduces the effects of slow processors on the estimated signal. Numerically, we demonstrate that this method can outperform other synchronous and asynchronous methods for solving SR.

Acknowledgements

This material is based upon work supported by the National Science Foundation under Grant No. ECCS-1810256, Grant No. CCF-1718195, Grant No. DMS-1522158, Award No. CAREER-1348721, Grant No. BIGDATA-1740325, and the University of California, Davis Dissertation Fellowship.

Contact Information

- https://www.math.ucla.edu/~jhaddock
- Email: jhaddock@math.ucla.edu

References

[1] A. Zaeemzadeh, J. Haddock, N. Rahnavard, and D. Needell.
A Bayesian approach for asynchronous parallel sparse recovery.
In *Proc. 52nd Asilomar Conference on Signals, Systems and Computers*, 2018.