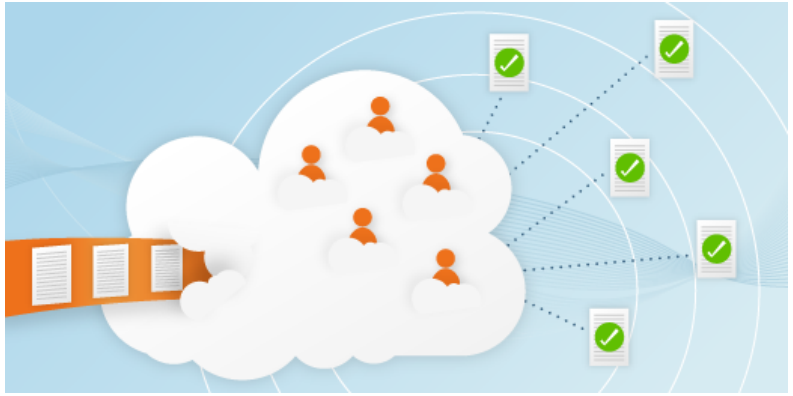
Message-passing Algorithms for Approximate Singular Vector Computation

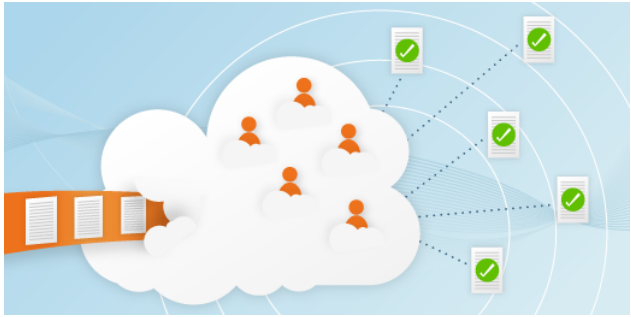
Sewoong Oh

Massachusetts Institute of Technology
joint work with David R. Karger(MIT) and Devavrat Shah(MIT)



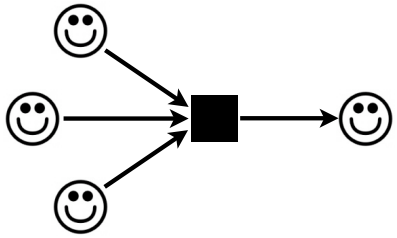
Outline

1. Inference Problem in Crowdsourcing

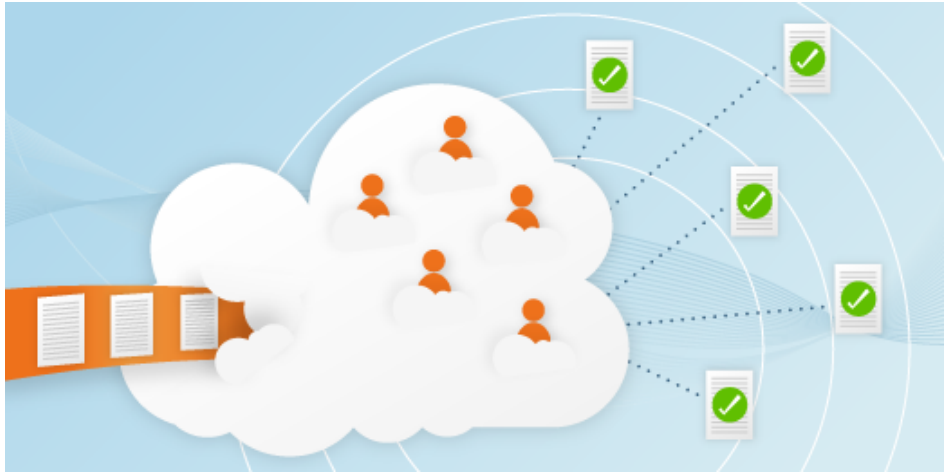


amazon mechanical turk™
Artificial Artificial Intelligence

2. Analysis of the Message-passing Algorithm

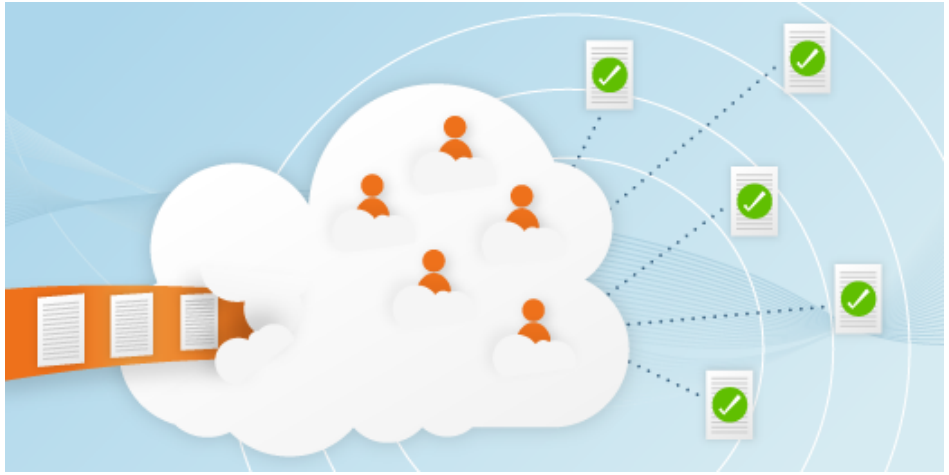


Crowdsourcing



	Speed/Cost	Reliability
Undergrad intern:	300 images/hr, cost: \$15/hr	95%
MTurk (single label):	3000 images/hr, cost: \$15/hr	65%

Crowdsourcing



	Speed/Cost	Reliability
Undergrad intern:	300 images/hr, cost: $15/hr	95%
MTurk (single label):	3000 images/hr, cost: $15/hr	65%
MTurk (multiple labels):	600 images/hr, cost: $15/hr	95%

Crowdsourcing system

Help find missing...

Help find missing...

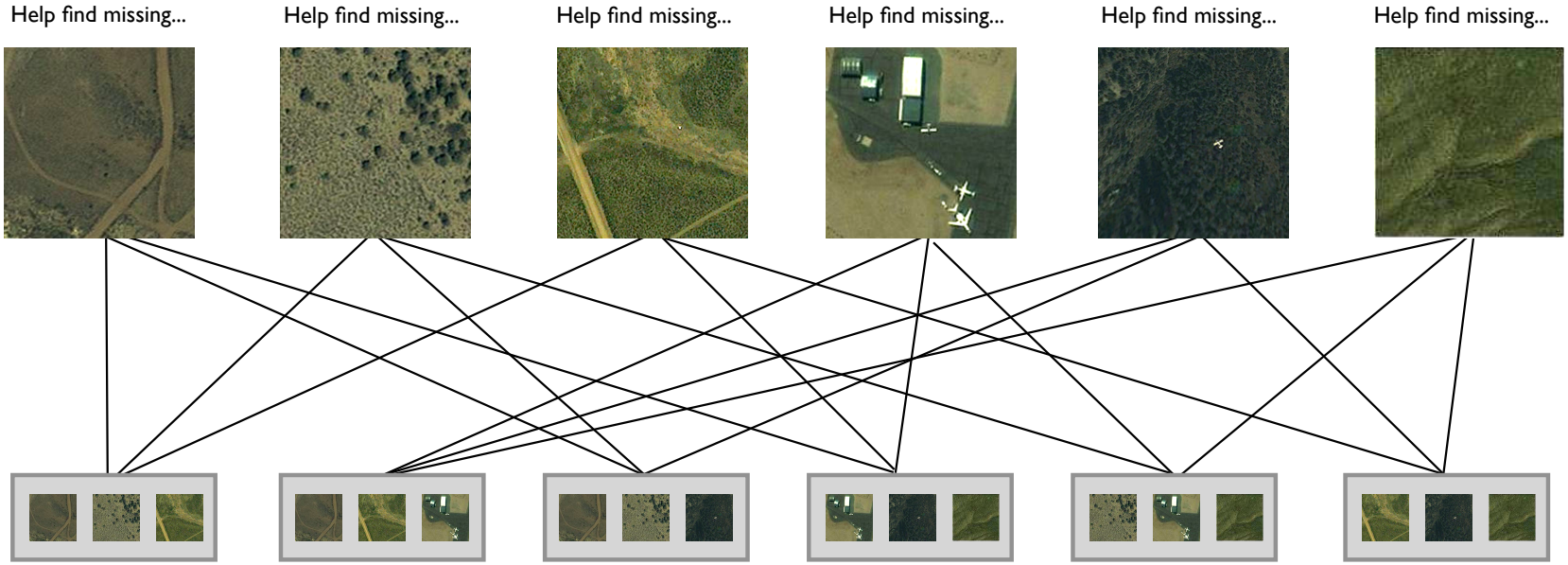
Help find missing...

Help find missing...

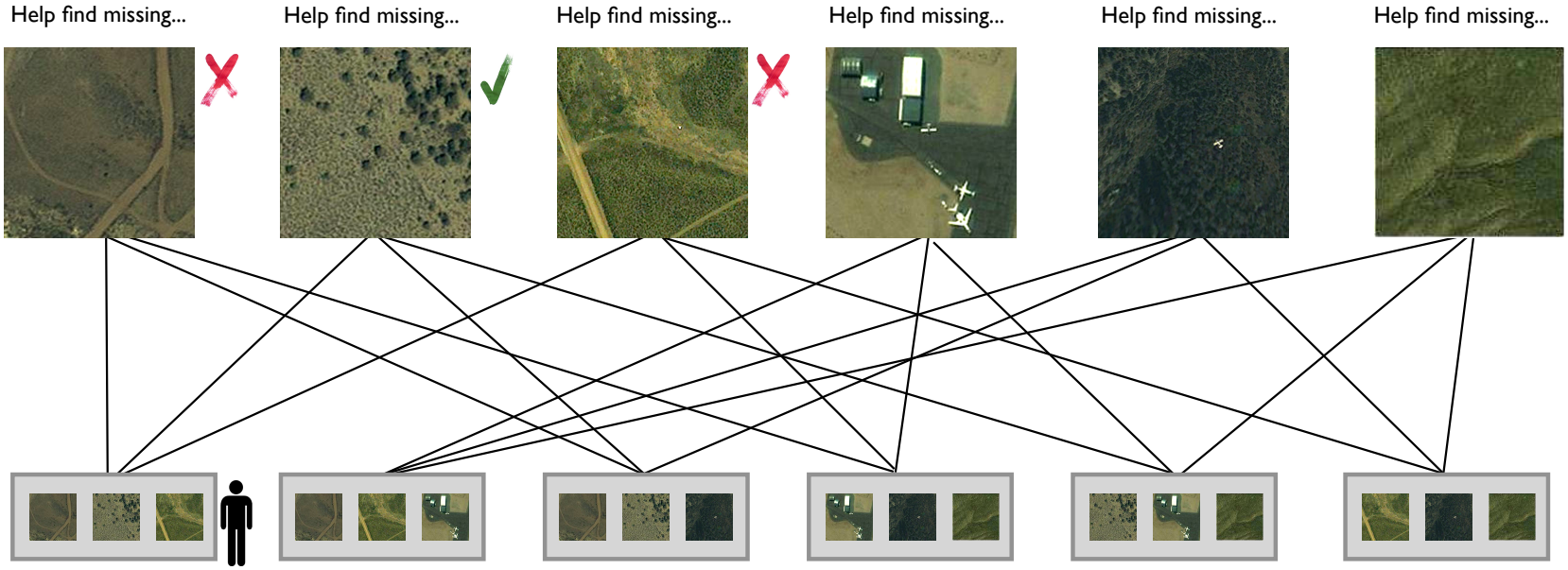
Help find missing...

Help find missing...

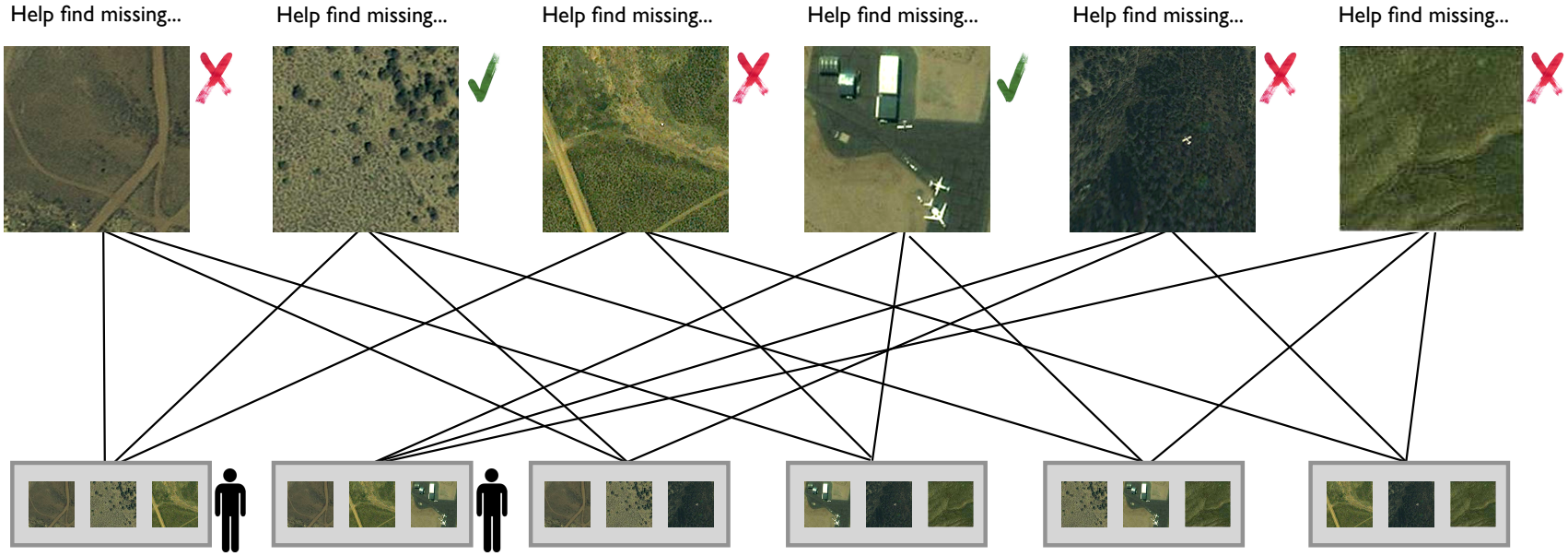
Crowdsourcing system



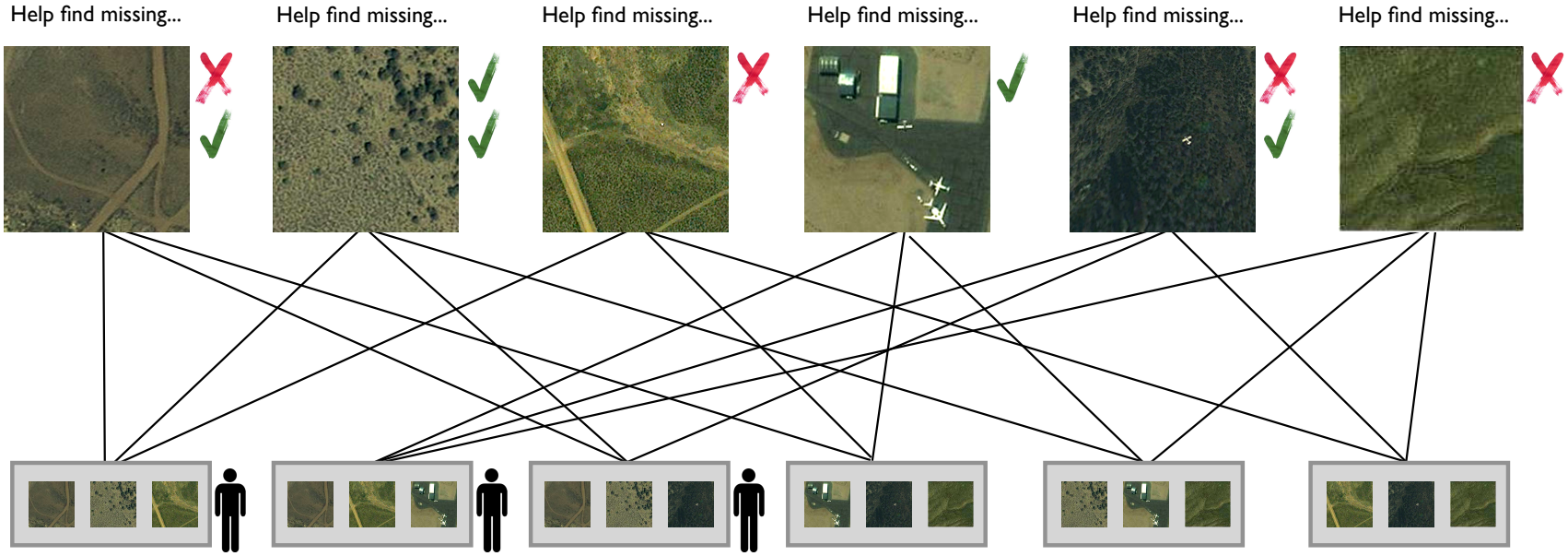
Crowdsourcing system



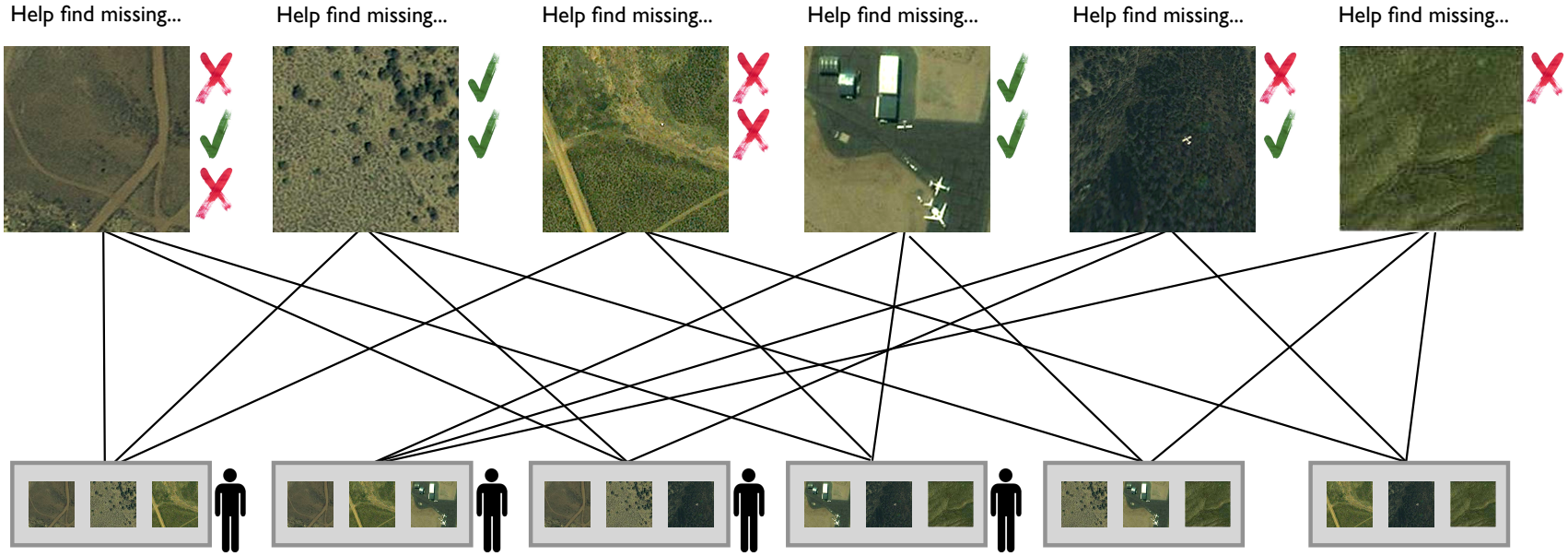
Crowdsourcing system



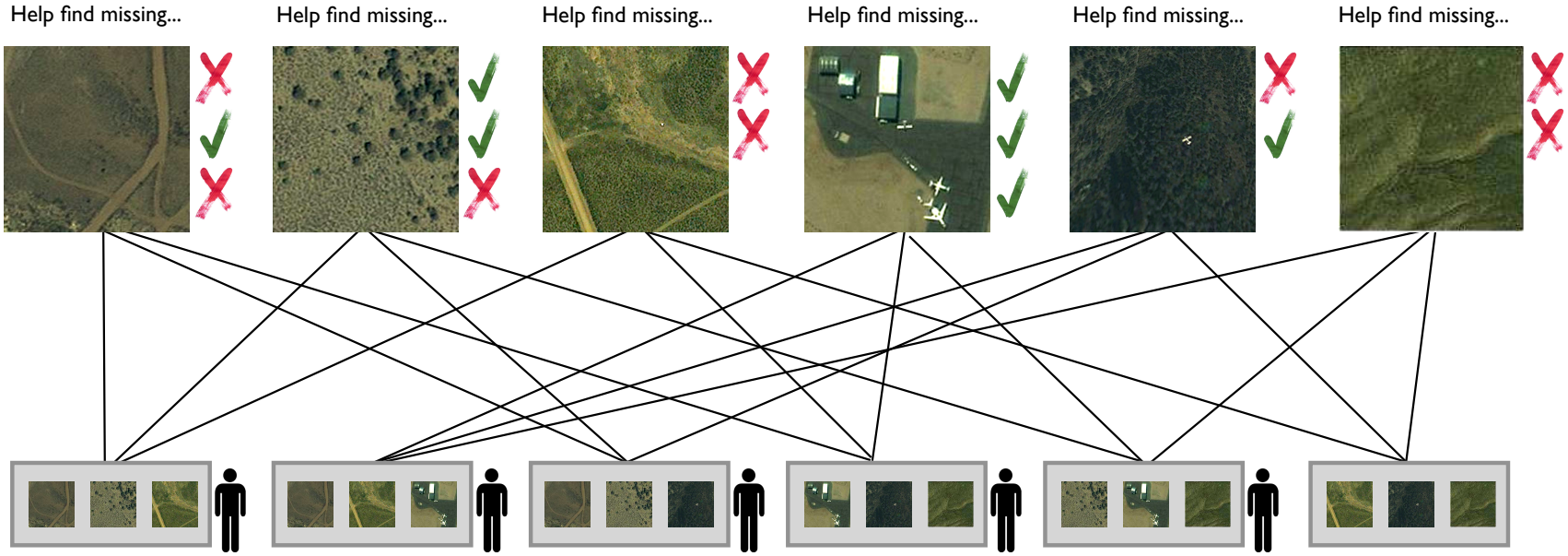
Crowdsourcing system



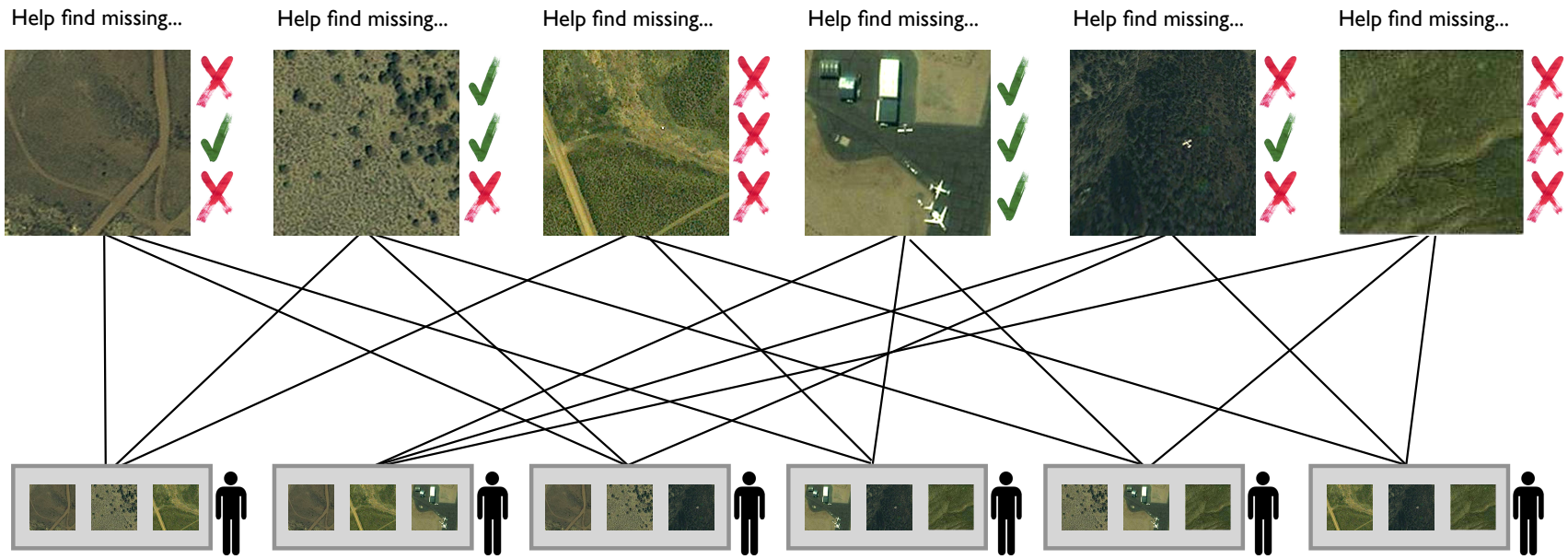
Crowdsourcing system



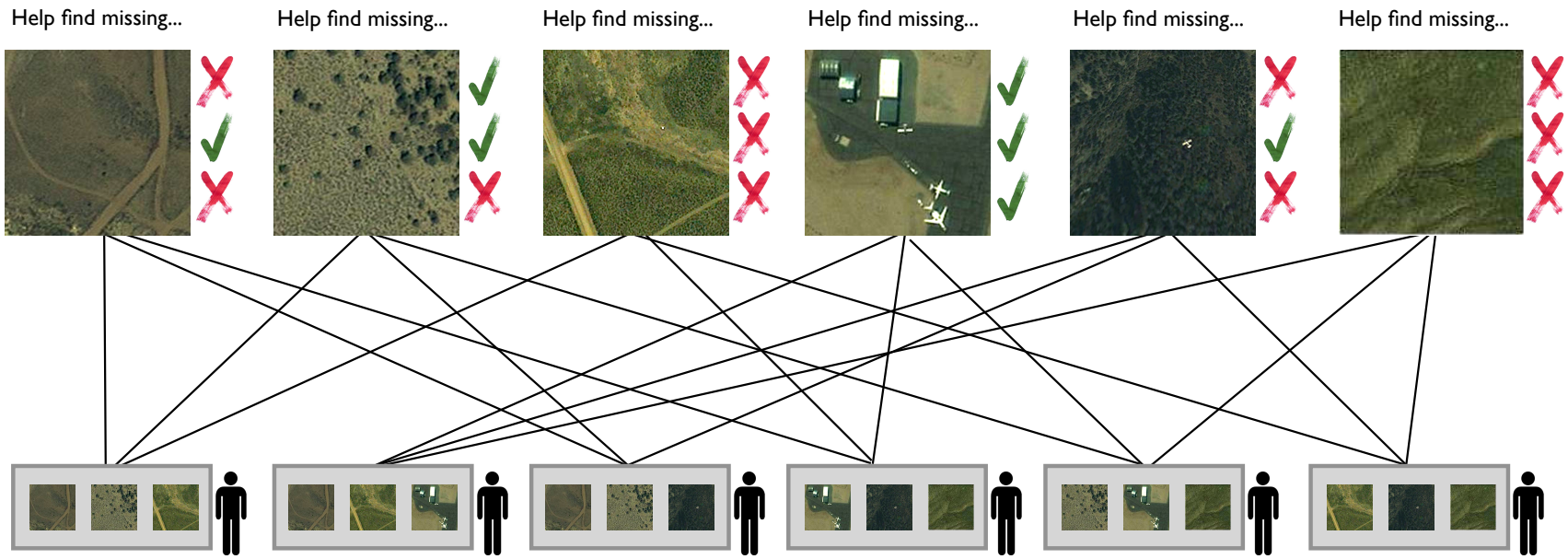
Crowdsourcing system



Crowdsourcing system



Crowdsourcing system



Goal: Reliably estimate the tasks at minimum cost

Challenges:

1. Task Assignment
2. Inference Problem

Previous approaches neglect the role of task assignment

1. Task Assignment — Which task should be assigned to which batch?
2. Inference Problem — What is the correct answer to the tasks?

Previous work

- Do not address the role of task assignment
- Focuses on inference problem given model
  - [Dawid,Skene '79] [Smyth et al. '95] [Sheng,Provost,Ipeirotis '08]
  - Expectation Maximization (EM) based heuristics
    - converges to local minimum
    - sensitive to initialization
    - no performance guarantees

Addressing both challenges together

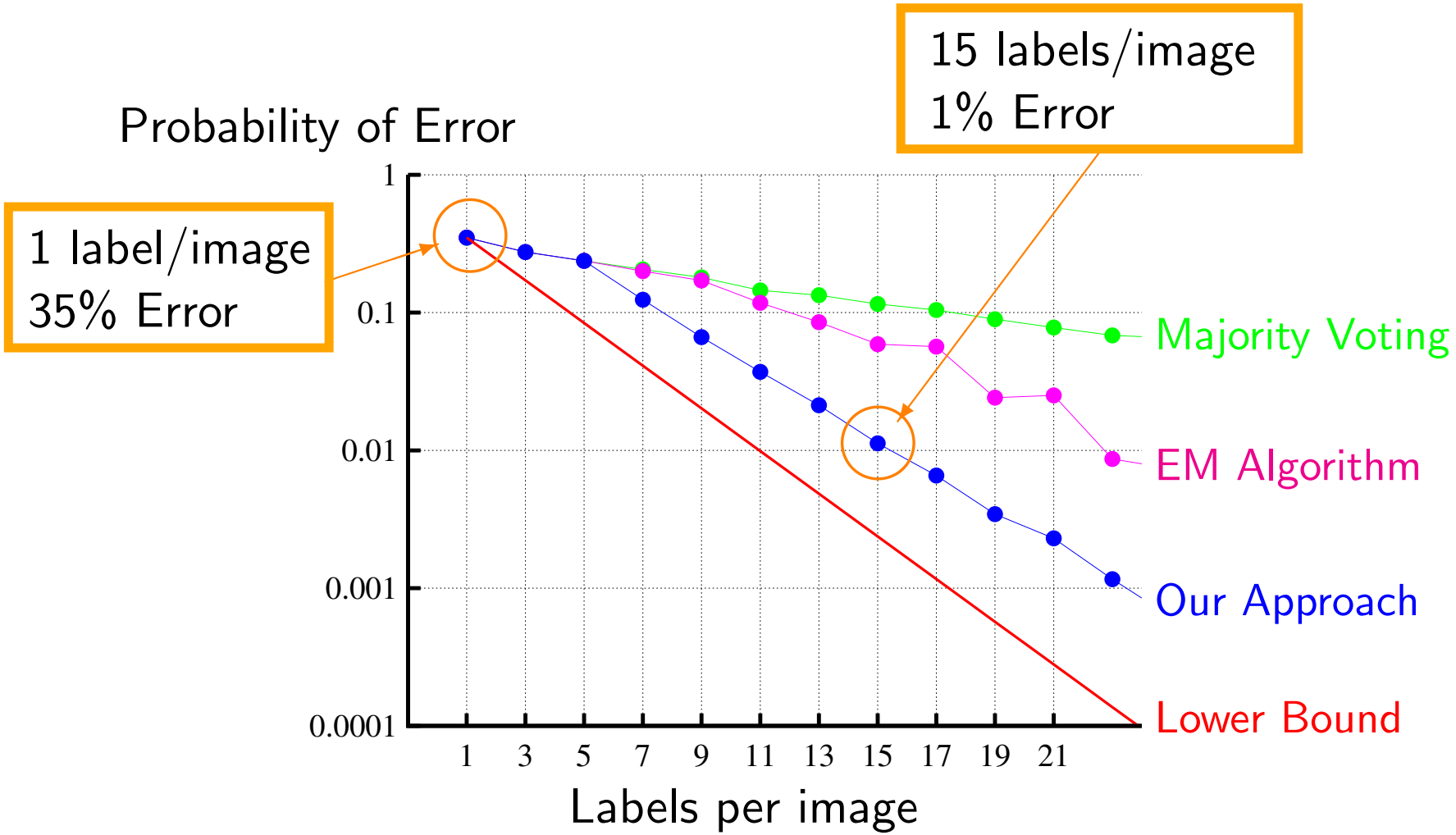
1. Task Assignment — Which task should be assigned to which batch?
2. Inference Problem — What is the correct answer to the tasks?

Our approach:

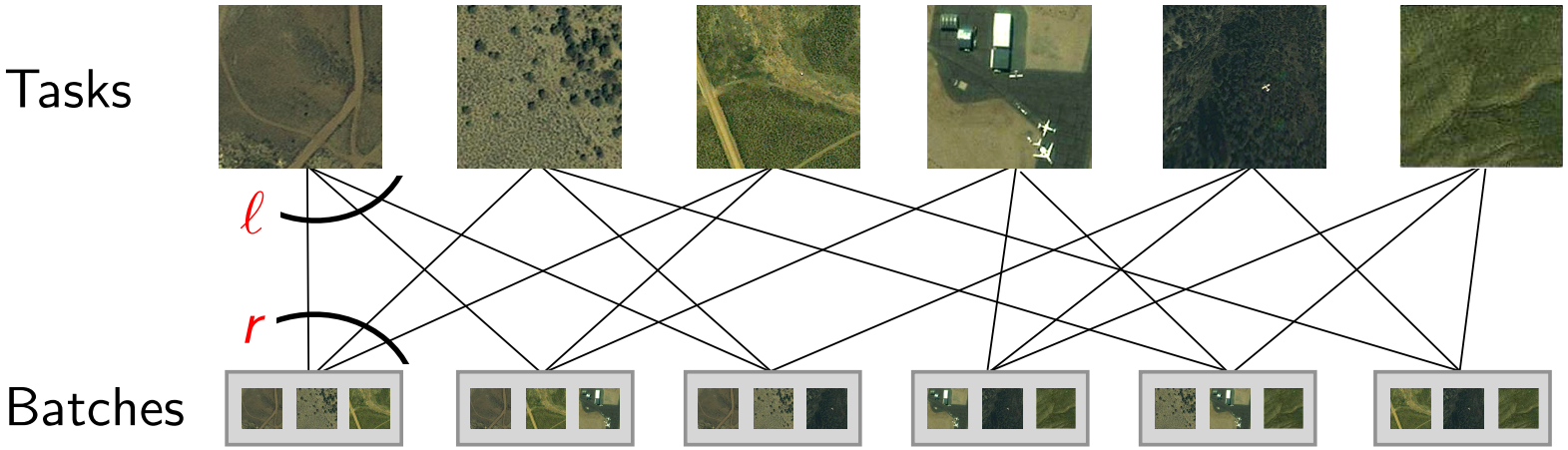
1. Task Assignment $\rightarrow$ Random Graph
2. Inference Problem $\rightarrow$ Low-rank Matrix Approximation

Extremely efficient and asymptotically order-optimal!

Preview of main result

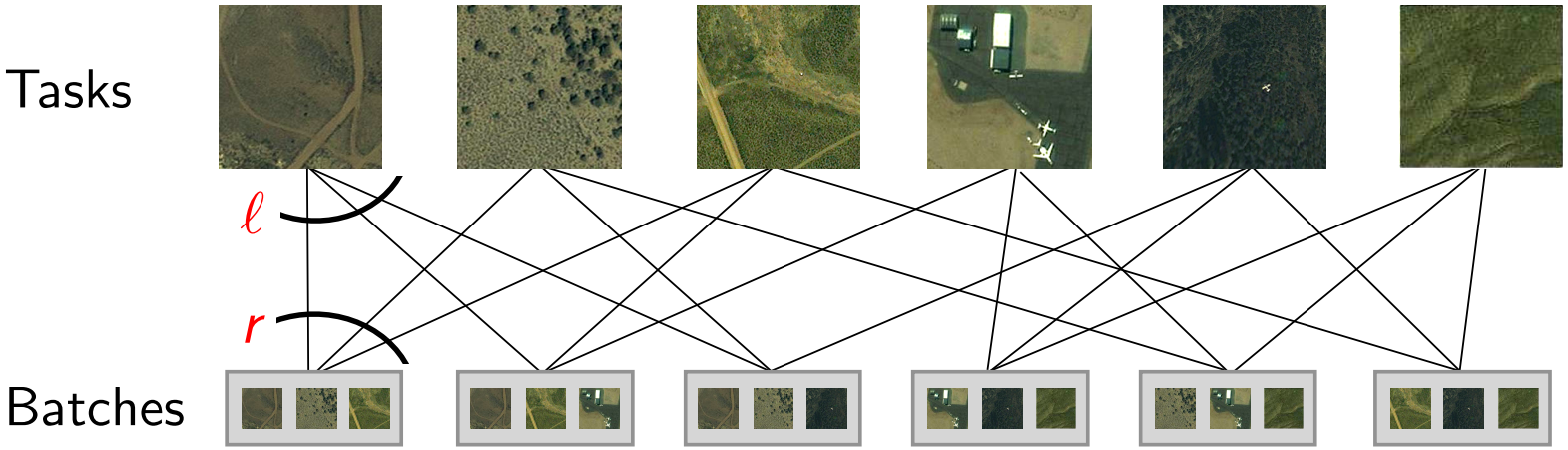


Task assignment

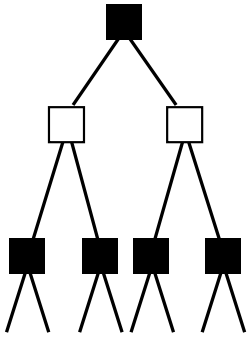


- Random (ℓ, r)-regular bipartite graphs

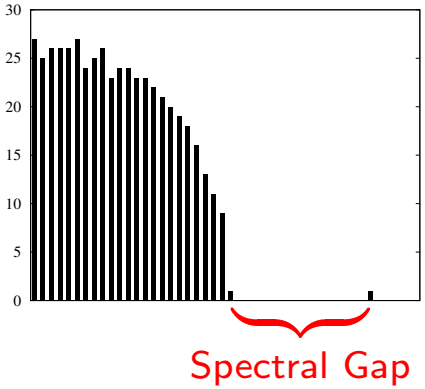
Task assignment



- Random (ℓ, r)-regular bipartite graphs have good properties
  - Locally Tree-like
  - Good Expander

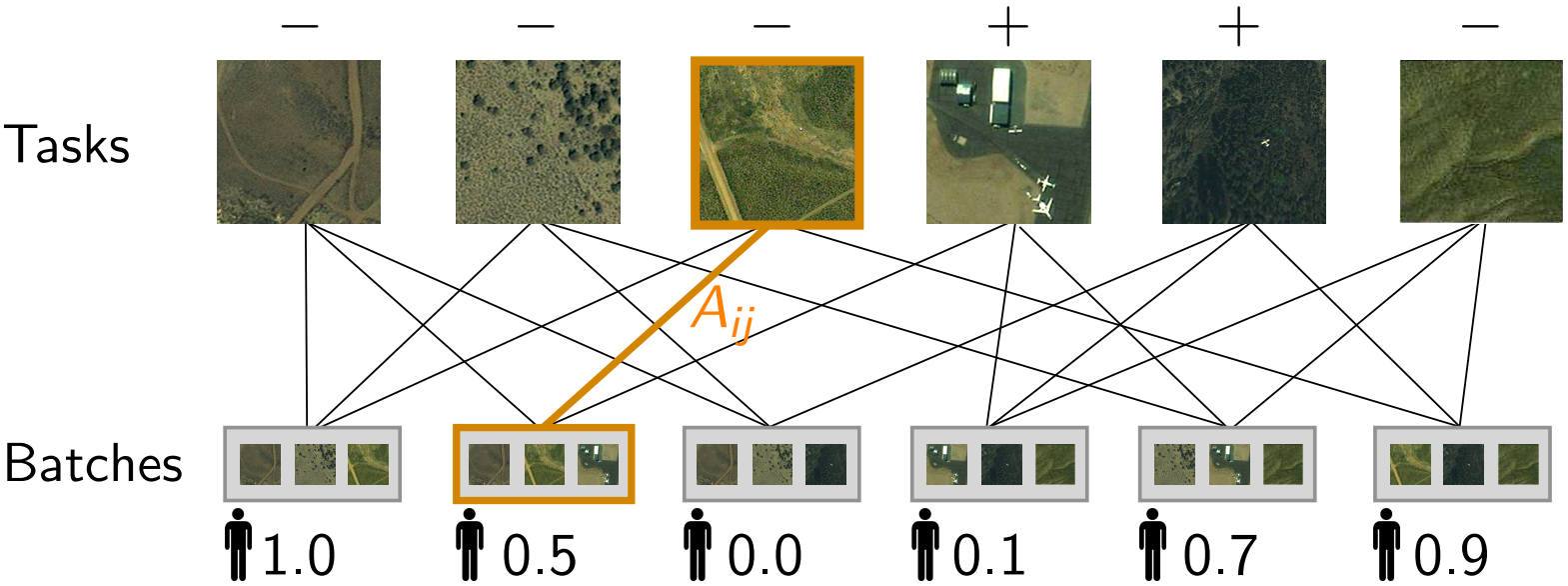


$\rightarrow$ Sharpen Analysis



$\rightarrow$ High Signal-to-Noise Ratio

Modeling the crowd

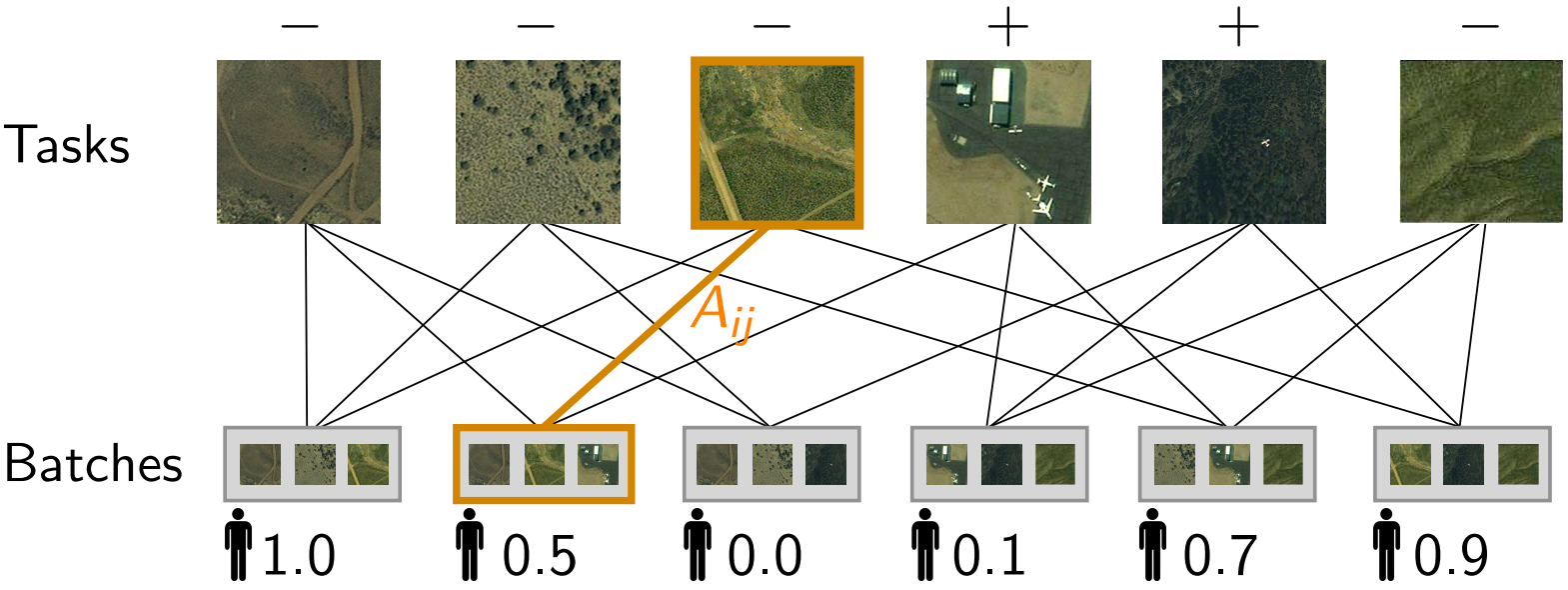


- Binary tasks: $t_i \in \{+1, -1\}$
- Worker reliability: $p_j \in [0, 1]$

$$A_{ij} = \begin{cases} t_i & \text{with probability } p_j \\ -t_i & \text{with probability } 1 - p_j \end{cases}$$

  A_{ij}'s are independent conditioned on t_i and p_j
- p_j drawn i.i.d. from $\mathcal{F}(p_j)$
- Necessary to assume we know whether $\mathbb{E}_{\mathcal{F}}[p_j] > 0.5$ or not

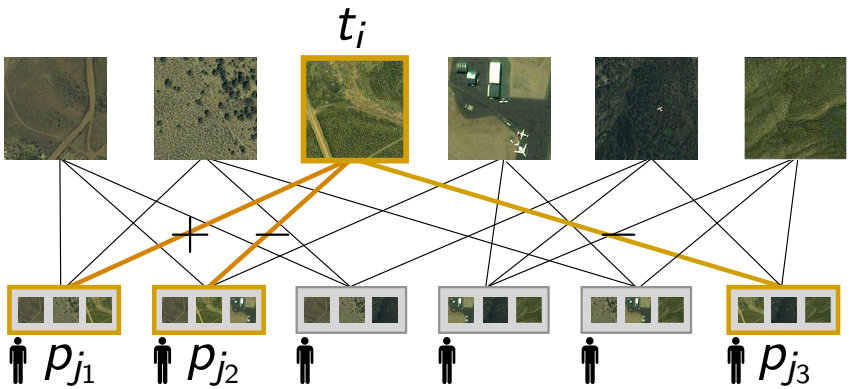
Modeling the crowd



- Underlying assumptions
  - Heterogeneous workers
  - Homogeneous tasks
  - Unbiased workers

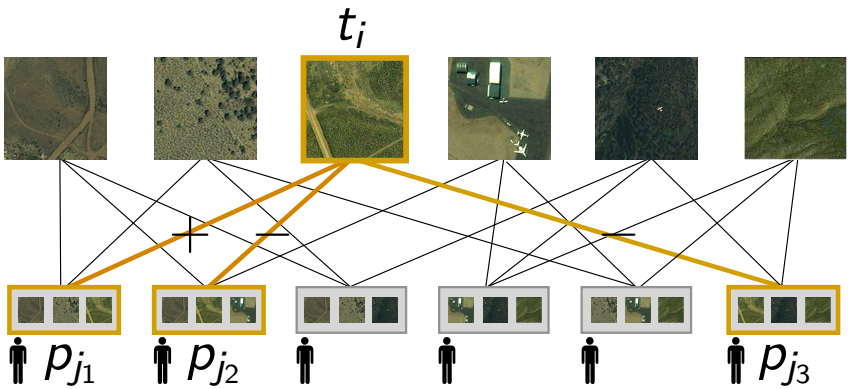
Inference problem

- Given the responses $\{A_{ij}\}$
- Infer the answers $\{t_i\}$



Inference problem

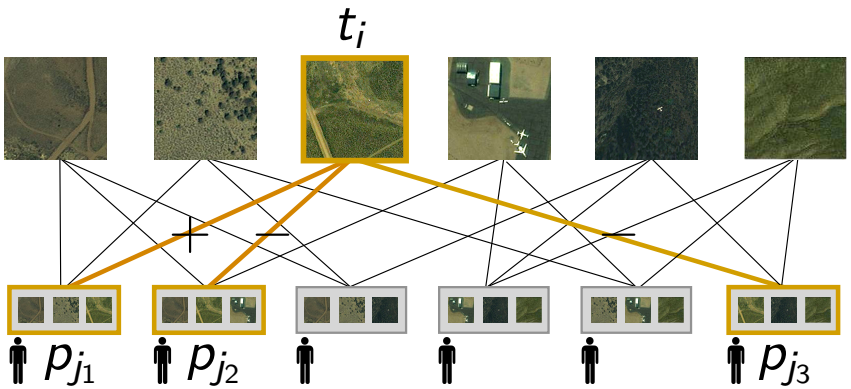
- Given the responses $\{A_{ij}\}$
- Infer the answers $\{t_i\}$



- Majority Voting:
  $\hat{t}_i = \mathrm{sign}\Big(\sum_j A_{ij}\Big)$

Inference problem

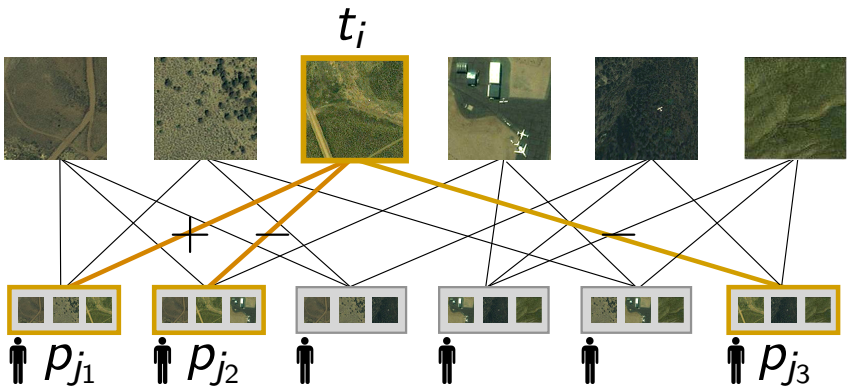
- Given the responses $\{A_{ij}\}$
- Infer the answers $\{t_i\}$



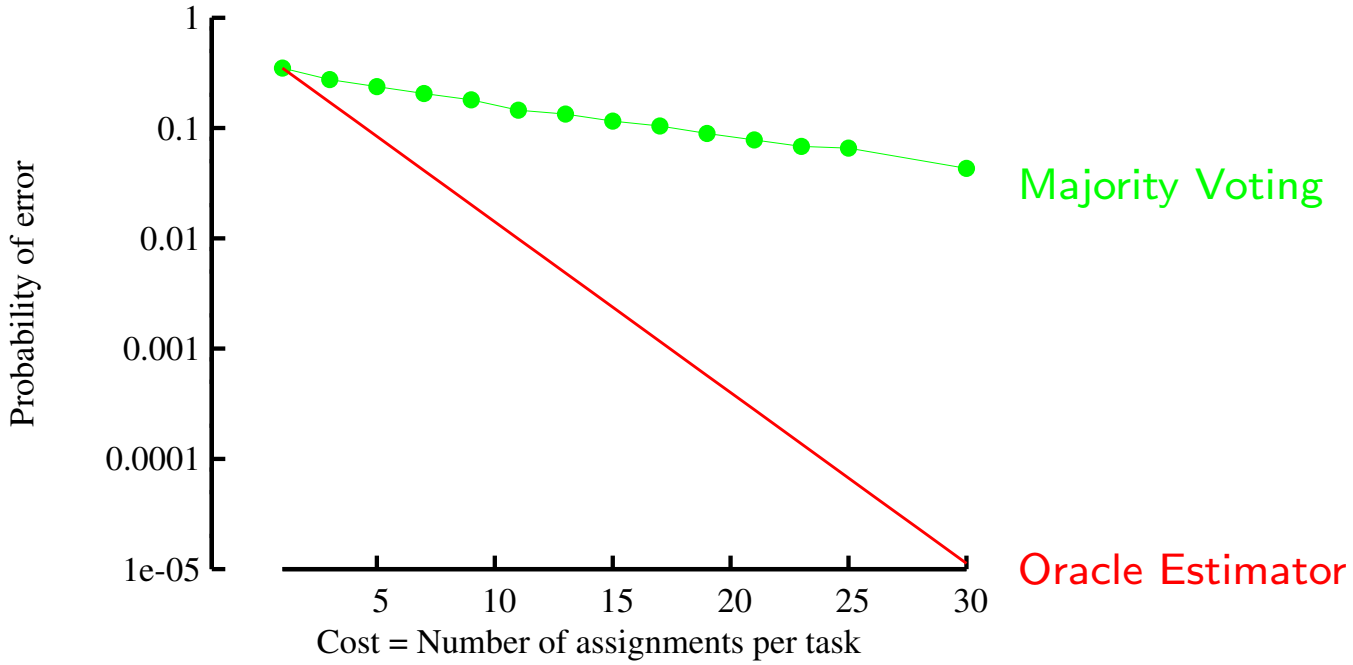
- Majority Voting:
  $$\hat{t}_i = \mathrm{sign}\Big(\sum_j A_{ij}\Big)$$
- Oracle Estimator who knows p_j's:
  $$\hat{t}_i = \mathrm{sign}\Big(\sum_j \log\Big(\frac{p_j}{1-p_j}\Big) A_{ij}\Big)$$

Inference problem

- Given the responses $\{A_{ij}\}$
- Infer the answers $\{t_i\}$

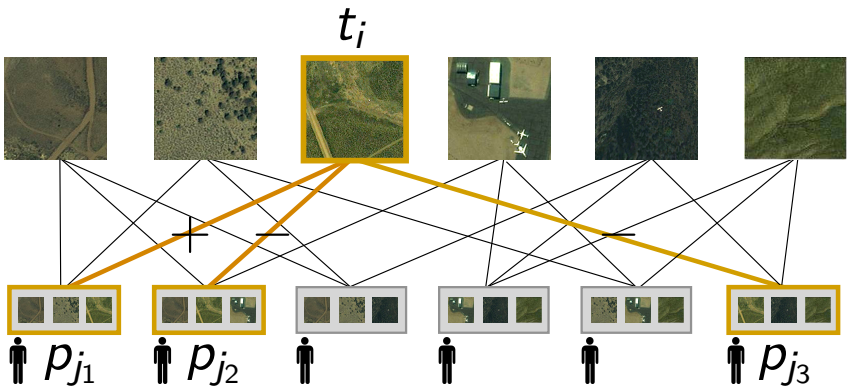


- Majority Voting:
  $$\hat{t}_i = \mathrm{sign}\Big(\sum_j A_{ij}\Big)$$
- Oracle Estimator who knows p_j's:
  $$\hat{t}_i = \mathrm{sign}\Big(\sum_j \log\Big(\frac{p_j}{1-p_j}\Big) A_{ij}\Big)$$

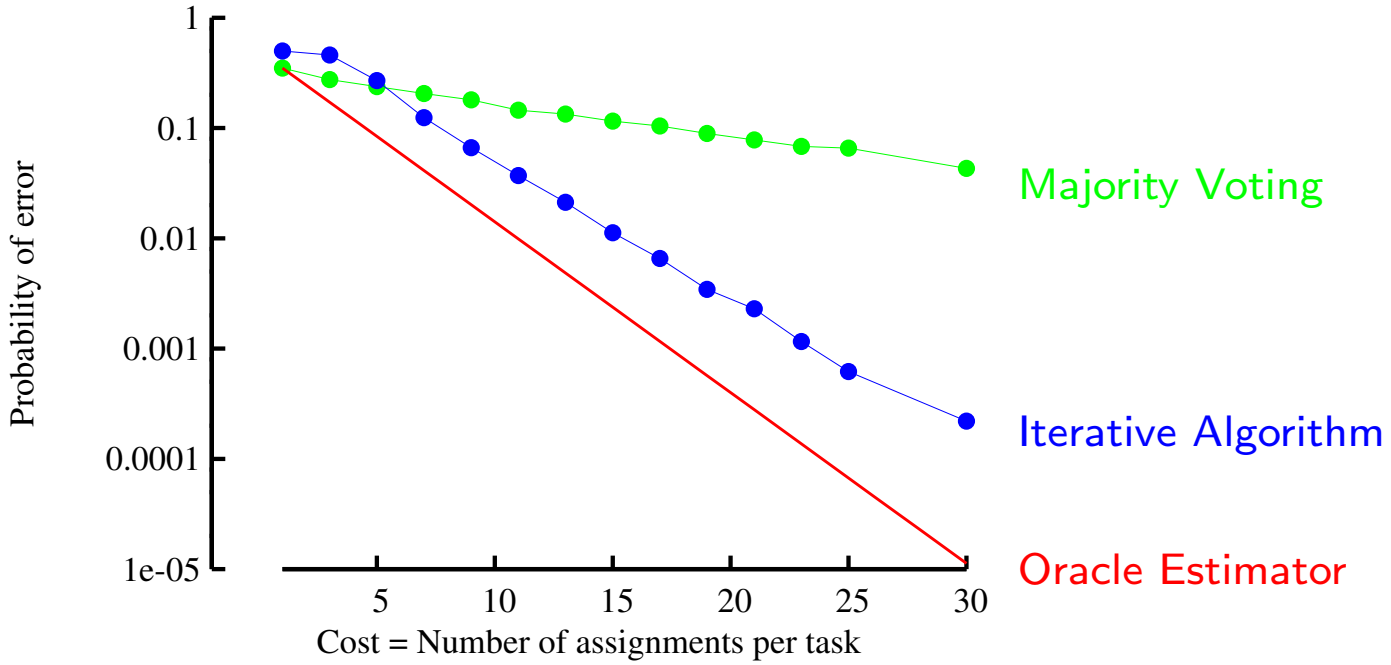


Inference problem

- Given the responses $\{A_{ij}\}$
- Infer the answers $\{t_i\}$

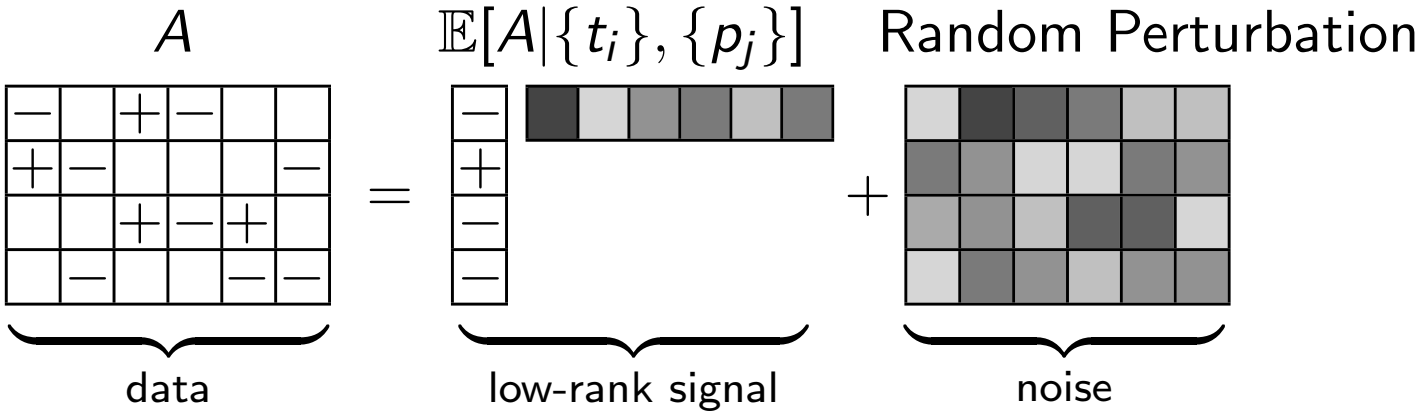


- Majority Voting:
  $\hat{t}_i = \mathrm{sign}\Big(\sum_j A_{ij}\Big)$
- Oracle Estimator who knows p_j's:
  $\hat{t}_i = \mathrm{sign}\Big(\sum_j \log\Big(\frac{p_j}{1-p_j}\Big) A_{ij}\Big)$
- Our Approach:
  Singular vectors



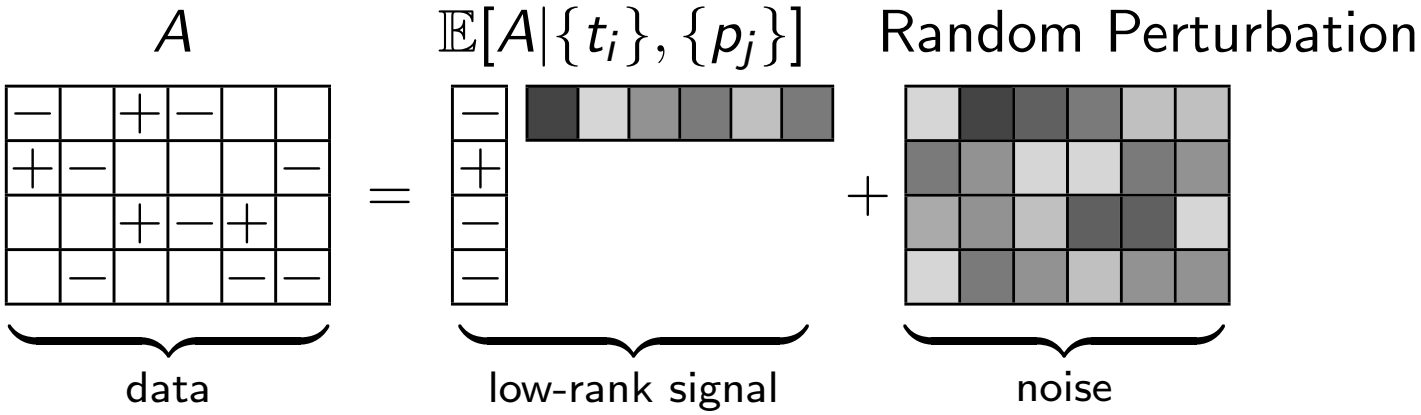
Inference using singular vectors

$$\mathbb{E}[A_{ij}|t_i, p_j] = t_i(2p_j - 1)$$



Inference using singular vectors

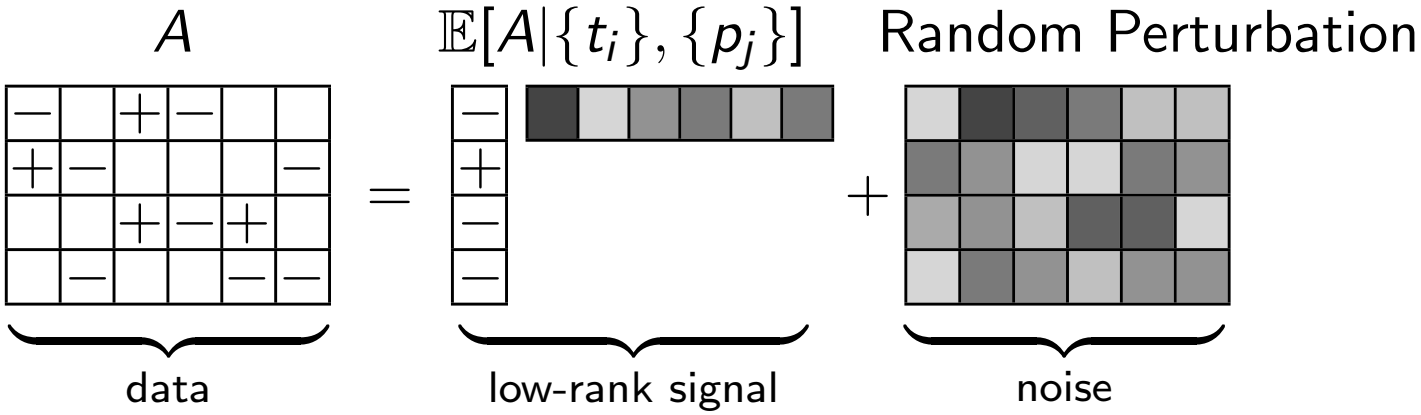
$$\mathbb{E}[A_{ij}|t_i, p_j] = t_i(2p_j - 1)$$



- Algorithm:
  1. Compute the first singular vectors (u, v) of A
  2. Output estimate: $\hat{t}_i = \mathrm{sign}(u_i)$

Inference using singular vectors

$$\mathbb{E}[A_{ij}|t_i, p_j] = t_i(2p_j - 1)$$



- Algorithm:
  1. Compute the first singular vectors (u, v) of A
  2. Output estimate: $\hat{t}_i = \mathrm{sign}(u_i)$

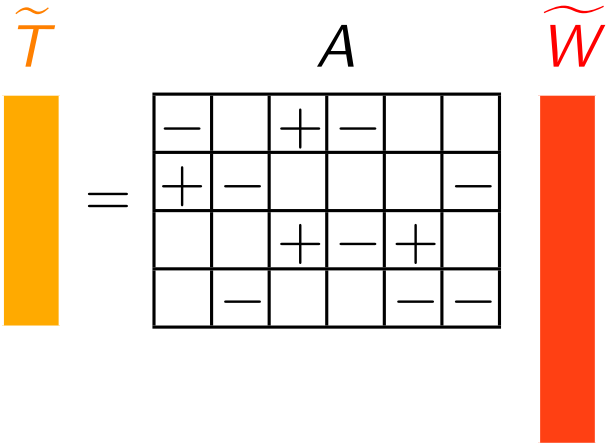
Problem

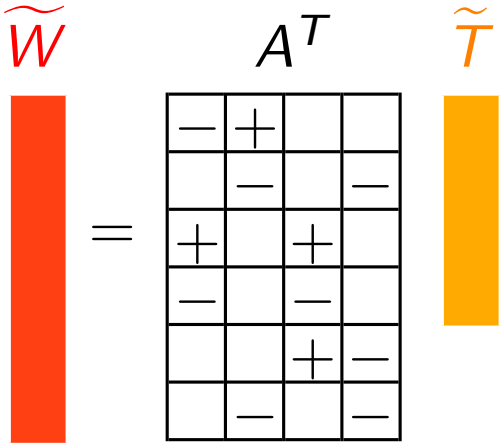
- [Achlioptas,McSherry 02], [Keshavan,Montanari,Oh 10]
- Current analysis techniques do not give tight bounds

Message-passing approach gives sharper analysis

- Power Iteration:

Iterate: $\widetilde{T}_i = \sum_{j \in \partial i} \widetilde{W}_j A_{ij}$, $\widetilde{W}_j = \sum_{i \in \partial j} \widetilde{T}_i A_{ij}$





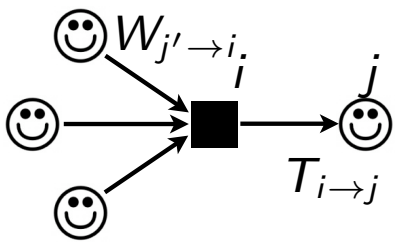
Message-passing approach gives sharper analysis

- Power Iteration:

Iterate: $\widetilde{T}_i = \sum_{j \in \partial i} \widetilde{W}_j A_{ij}$, $\widetilde{W}_j = \sum_{i \in \partial j} \widetilde{T}_i A_{ij}$

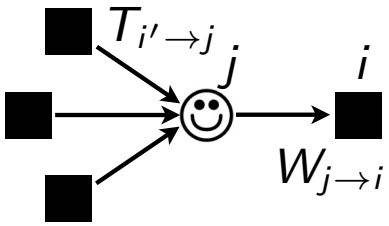
- Message-passing Power Iteration:
  Iteratively update messages $\{T_{i \to j}\}$ and $\{W_{j \to i}\}$

Task-likelihood update



$$T_{i \to j} = \sum_{j' \in \partial i \setminus j} W_{j' \to i} A_{ij'}$$

Worker-reliability update

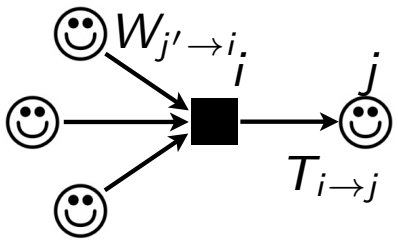


$$W_{j \to i} = \sum_{i' \in \partial j \setminus i} T_{i' \to j} A_{i'j}$$

Message-passing approach gives sharper analysis

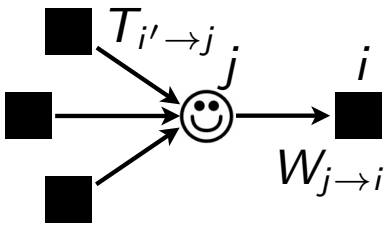
- Message-passing Power Iteration:

Task-likelihood update



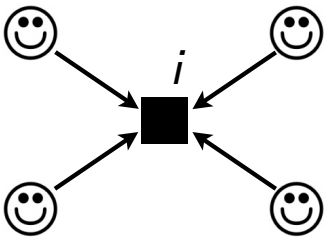
$$T_{i \to j} = \sum_{j' \in \partial i \setminus j} W_{j' \to i} A_{ij'}$$

Worker-reliability update



$$W_{j \to i} = \sum_{i' \in \partial j \setminus i} T_{i' \to j} A_{i'j}$$

Decision update



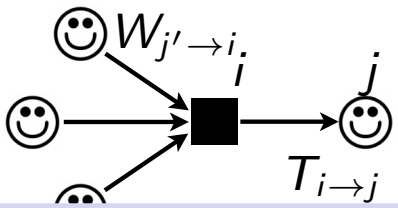
$$T_i = \sum_{j \in \partial i} W_{j \to i} A_{ij}$$

$$\hat{t}_i = \mathrm{sign}(T_i)$$

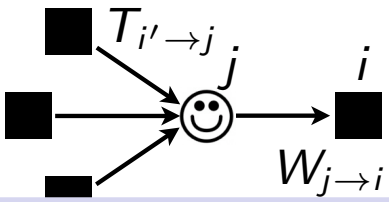
Message-passing approach gives sharper analysis

- Message-passing Power Iteration:

Task-likelihood update



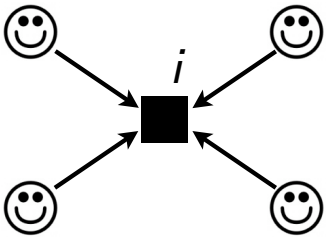
Worker-reliability update



Our Inference Algorithm

- Requires no knowledge of $\mathcal{F}$
- Provide sharp bound on the probability of error

Decision update

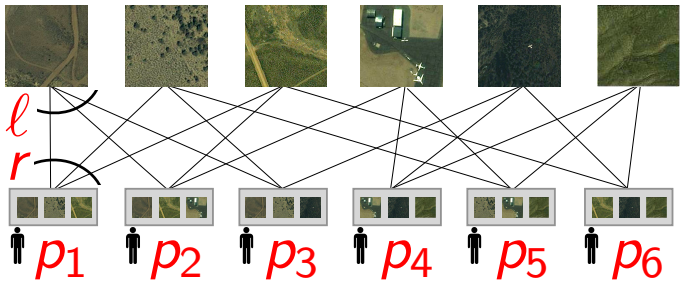


$$T_i = \sum_{j \in \partial i} W_{j \to i} A_{ij}$$

$$\hat{t}_i = \mathrm{sign}(T_i)$$

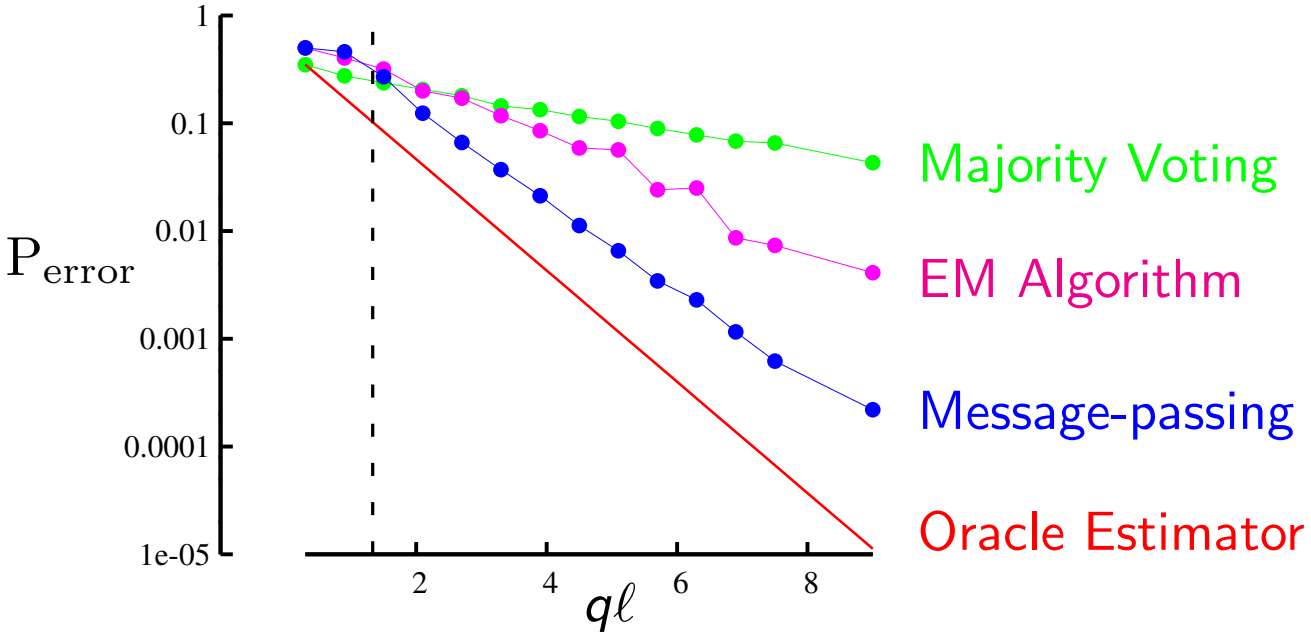
Performance Analysis

Performance depends on collective quality of the crowd



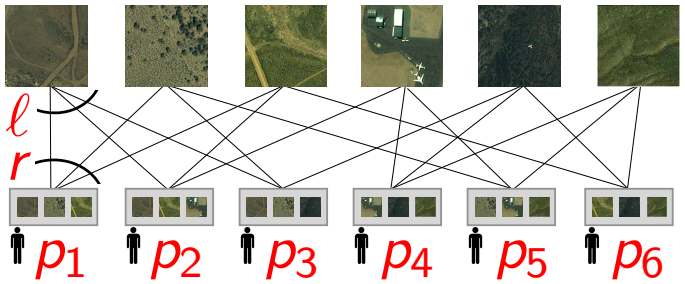
Probability of error depends on $\mathcal{F}(\cdot)$ through Crowd Quality Parameter:

$$q \equiv \mathbb{E}_{\mathcal{F}}[(2p_j - 1)^2]$$



- Phase transition at $q^2(\ell - 1)(r - 1) = 1$
  - $q^2(\ell - 1)(r - 1) < 1$: Iteration Hurts
  - $q^2(\ell - 1)(r - 1) > 1$: Iteration Helps

Sharp bound on the probability of error



- Probability of error depends on $\mathcal{F}(\cdot)$ through:

$$q \equiv \mathbb{E}_{\mathcal{F}}[(2p_j - 1)^2]$$

Theorem. [Karger, O., Shah '11]

In the large system limit, and for $(\ell - 1)(r - 1)q^2 > 1$, using a random graph and k iterations of the message-passing algorithm achieves

$$\lim_{m\to\infty} \frac{1}{m} \sum_{i=1}^{m} \mathbb{P}(t_i \neq \hat{t}_i) \leq \exp\left\{ -\frac{q\ell}{2\sigma_k^2} \right\},$$

for $\sigma_k^2 \equiv \left(3 + \frac{1}{qr}\right) \frac{q^2(\ell-1)(r-1)}{q^2(\ell-1)(r-1)-1} + \frac{2}{q}(q^2(\ell-1)(r-1))^{-k}$.

Upper bound matches the minimax lower bound

- Message-passing algorithm ($\ell \leq r$, $q(\ell-1) > 1$, $k \to \infty$):

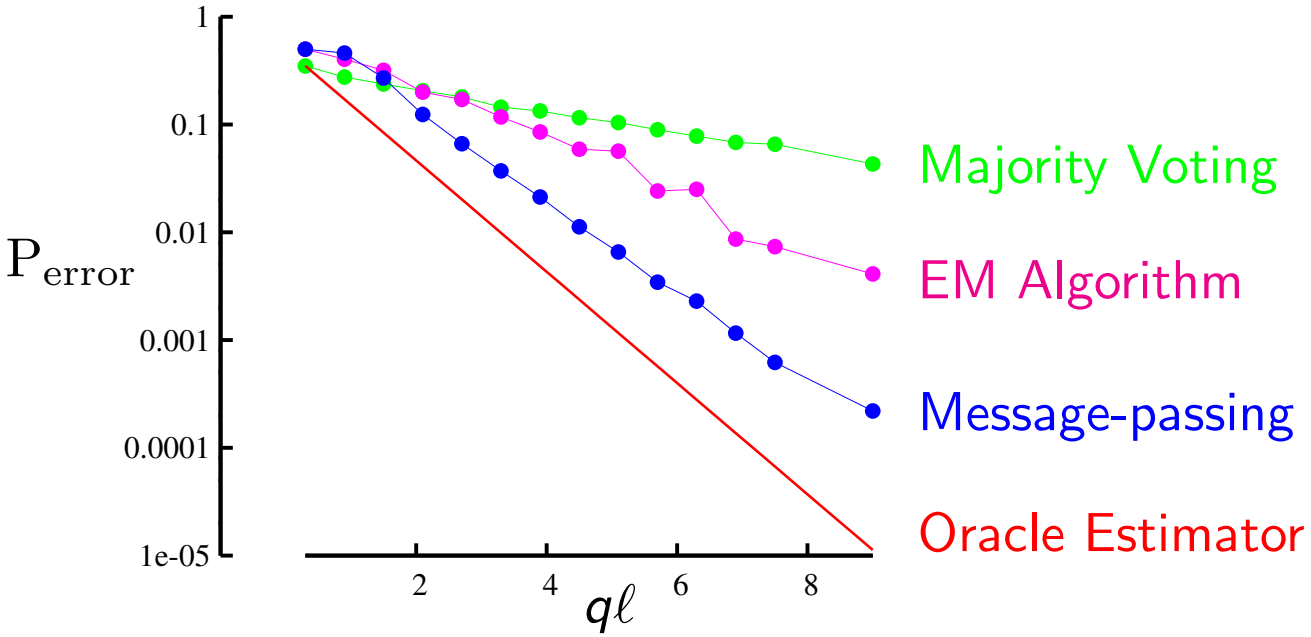
$$\mathrm{P}_{\mathrm{error}} \leq e^{-\frac{1}{8}(q\ell-1)}$$

- Matching minimax lower bound:

$$\inf_{\mathrm{Alg}, G(\ell)} \sup_{\{t_i\}, \{\mathcal{F} | \mathbb{E}[(2p-1)^2]=q\}} \mathrm{P}_{\mathrm{error}} \gtrsim \frac{1}{2} e^{-(q\ell + O(q^2\ell))}$$

- Majority Voting:

$$\inf_{G(\ell)} \sup_{\{t_i\}, \{\mathcal{F} | \mathbb{E}[(2p-1)^2]=q\}} \mathrm{P}_{\mathrm{error}} \geq e^{-C(q^2\ell+1)}$$



Our approach is asymptotically order-optimal

$$\mathrm{P}_{\mathrm{error}} \leq e^{-\frac{1}{8}(q\ell-1)}$$

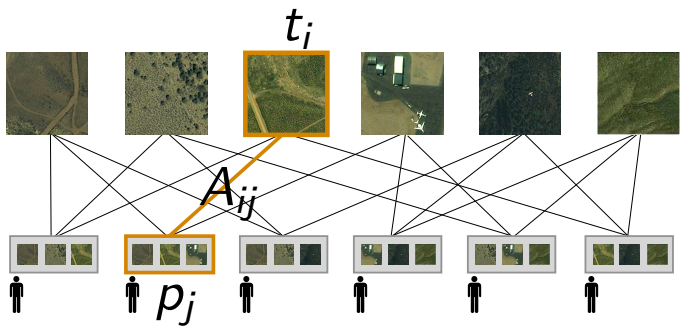
- How much do we need to spend to achieve $\mathrm{P}_{\mathrm{error}} \leq \epsilon$?
  - Sufficient to choose $\ell = O\left(\frac{1}{q}\log(\frac{1}{\epsilon})\right)$
  - Necessary to have $\ell = \Omega\left(\frac{1}{q}\log(\frac{1}{\epsilon})\right)$

Proof Sketch

Recap

- Graphical model
  - task variables: $t_i \in \{\pm 1\}$
  - worker variables: i.i.d. $p_j \sim \mathcal{F}(\cdot)$

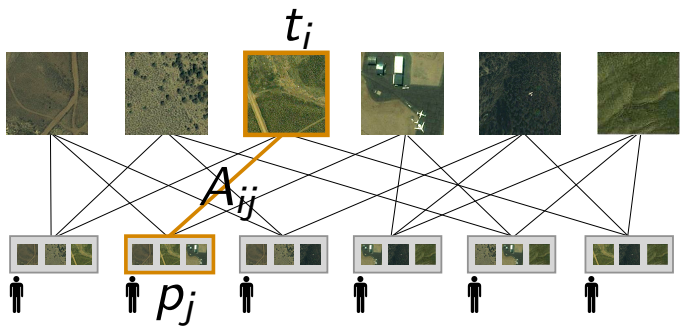
$$A_{ij} = \begin{cases} t_i & \text{with probability } p_j \\ -t_i & \text{with probability } 1 - p_j \end{cases}$$



Recap

- Graphical model
  - task variables: $t_i \in \{\pm 1\}$
  - worker variables: i.i.d. $p_j \sim \mathcal{F}(\cdot)$

$$A_{ij} = \begin{cases} t_i & \text{with probability } p_j \\ -t_i & \text{with probability } 1 - p_j \end{cases}$$



- Message-passing Power Iteration:

Iterate

$$T_{i\to j}^{(k)} = \sum_{j' \in \partial i \setminus j} W_{j'\to i}^{(k-1)} A_{ij'}$$

$$W_{j\to i}^{(k)} = \sum_{i' \in \partial j \setminus i} T_{i'\to j}^{(k)} A_{i'j}$$

$\Longrightarrow$

Final Estimate

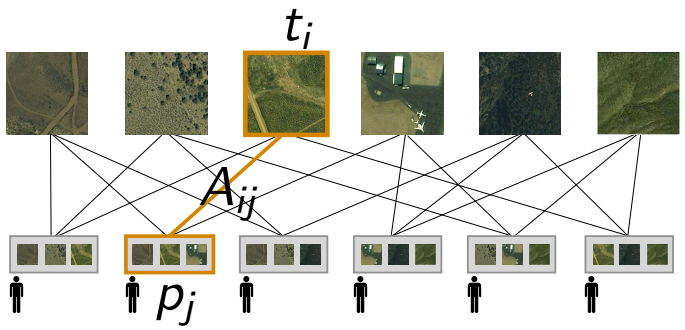
$$T_i^{(k)} = \sum_{j \in \partial i} W_{j\to i}^{(k-1)} A_{ij}$$

$$\hat{t}_i^{(k)} = \mathrm{sign}(T_i^{(k)})$$

Recap

- Graphical model
  - task variables: $t_i \in \{\pm 1\}$
  - worker variables: i.i.d. $p_j \sim \mathcal{F}(\cdot)$

$$A_{ij} = \begin{cases} t_i & \text{with probability } p_j \\ -t_i & \text{with probability } 1 - p_j \end{cases}$$



- Message-passing Power Iteration:

Iterate

$$T_{i\to j}^{(k)} = \sum_{j' \in \partial i \setminus j} W_{j'\to i}^{(k-1)} A_{ij'}$$

$$W_{j\to i}^{(k)} = \sum_{i' \in \partial j \setminus i} T_{i'\to j}^{(k)} A_{i'j}$$

$\Longrightarrow$

Final Estimate

$$T_i^{(k)} = \sum_{j \in \partial i} W_{j\to i}^{(k-1)} A_{ij}$$

$$\hat{t}_i^{(k)} = \mathrm{sign}(T_i^{(k)})$$

- Performance metric: (by symmetry, assume all $t_i = +1$)

$$\lim_{m\to\infty} \frac{1}{m} \sum_{i=1}^{m} \mathbb{P}(\hat{t}_i^{(k)} = -1) = \lim_{m\to\infty} \frac{1}{m} \sum_{i=1}^{m} \mathbb{P}(T_i^{(k)} \le 0)$$

	Message Update	Density Evolution
Initialize:	$W_{j\to i}^{(0)} \sim \mathcal{N}(1,1)$	$\mathbf{W}^{(0)} \sim \mathcal{N}(1,1)$

	Message Update	Density Evolution
Initialize:	$W_{j\to i}^{(0)} \sim \mathcal{N}(1,1)$	$\mathbf{W}^{(0)} \sim \mathcal{N}(1,1)$
1st step:	$T_{i\to j}^{(1)} = \sum_{j'\in\partial i\setminus j} W_{j'\to i}^{(0)} A_{ij'}$	$\mathbf{T}^{(1)} \stackrel{d}{=} \sum_{j=1}^{\ell-1} \mathbf{W}_j^{(0)} \mathbf{A}_{\mathbf{p}_j,j}$
	$W_{j\to i}^{(1)} = \sum_{i'\in\partial j\setminus i} T_{i'\to j}^{(1)} A_{i'j}$	$\mathbf{W}_p^{(1)} \stackrel{d}{=} \sum_{i=1}^{r-1} \mathbf{T}_i^{(1)} \mathbf{A}_{p,i}$

$$\mathbf{A}_p = \begin{cases} +1 & \text{w.p. } p \\ -1 & \text{w.p. } 1-p \end{cases}$$

$$\mathbf{p} \sim \mathcal{F}$$

	Message Update	Density Evolution
Initialize:	$W_{j\to i}^{(0)} \sim \mathcal{N}(1,1)$	$\mathbf{W}^{(0)} \sim \mathcal{N}(1,1)$
k-th step:	$T_{i\to j}^{(k)} = \sum_{j'\in\partial i\setminus j} W_{j'\to i}^{(k-1)} A_{ij'}$	$\mathbf{T}^{(k)} \stackrel{d}{=} \sum_{j=1}^{\ell-1} \mathbf{W}_{\mathbf{p}_j,j}^{(k-1)} \mathbf{A}_{\mathbf{p}_j,j}$
	$W_{j\to i}^{(k)} = \sum_{i'\in\partial j\setminus i} T_{i'\to j}^{(k)} A_{i'j}$	$\mathbf{W}_{p}^{(k)} \stackrel{d}{=} \sum_{i=1}^{r-1} \mathbf{T}_{i}^{(k)} \mathbf{A}_{p,i}$
	$T_{i}^{(k)} = \sum_{j'\in\partial i} W_{j'\to i}^{(k-1)} A_{ij'}$	$\widehat{\mathbf{T}}^{(k)} \stackrel{d}{=} \sum_{j=1}^{\ell} \mathbf{W}_{\mathbf{p}_j,j}^{(k-1)} \mathbf{A}_{\mathbf{p}_j,j}$

	Message Update	Density Evolution
Initialize:	$W_{j\to i}^{(0)} \sim \mathcal{N}(1,1)$	$\mathbf{W}^{(0)} \sim \mathcal{N}(1,1)$
k-th step:	$T_{i\to j}^{(k)} = \sum_{j'\in\partial i\setminus j} W_{j'\to i}^{(k-1)} A_{ij'}$	$\mathbf{T}^{(k)} \stackrel{d}{=} \sum_{j=1}^{\ell-1} \mathbf{W}_{\mathbf{p}_j,j}^{(k-1)} \mathbf{A}_{\mathbf{p}_j,j}$
	$W_{j\to i}^{(k)} = \sum_{i'\in\partial j\setminus i} T_{i'\to j}^{(k)} A_{i'j}$	$\mathbf{W}_{p}^{(k)} \stackrel{d}{=} \sum_{i=1}^{r-1} \mathbf{T}_{i}^{(k)} \mathbf{A}_{p,i}$
	$T_{i}^{(k)} = \sum_{j'\in\partial i} W_{j'\to i}^{(k-1)} A_{ij'}$	$\widehat{\mathbf{T}}^{(k)} \stackrel{d}{=} \sum_{j=1}^{\ell} \mathbf{W}_{\mathbf{p}_j,j}^{(k-1)} \mathbf{A}_{\mathbf{p}_j,j}$

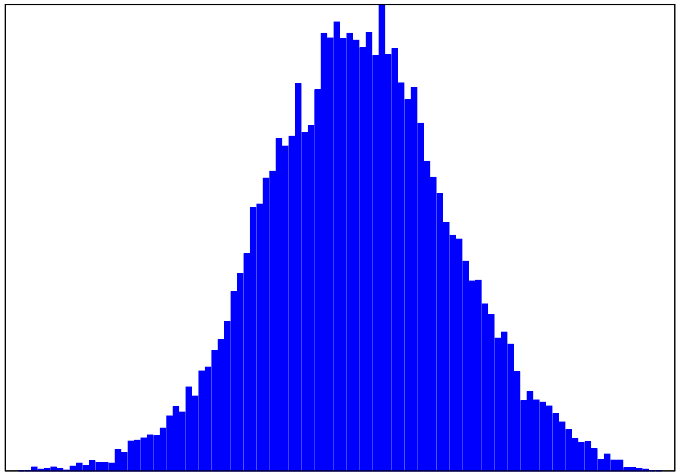
Challenges

- Random variables take real values
- Cannot describe/track the probability density functions
- Cannot evaluate $P_{\text{error}} = \mathbb{P}\big(\widehat{\mathbf{T}}^{(k)} \leq 0\big)$

Find a simple structure that describes the messages

- Histogram of $T_{i \to j}$'s

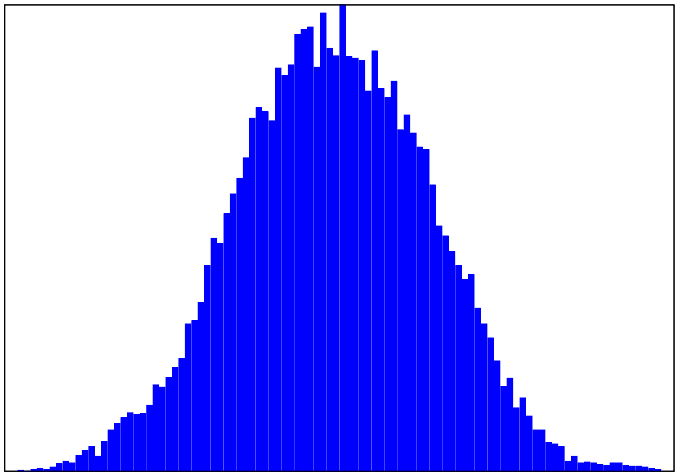
1st iteration



Find a simple structure that describes the messages

- Histogram of $T_{i \to j}$'s

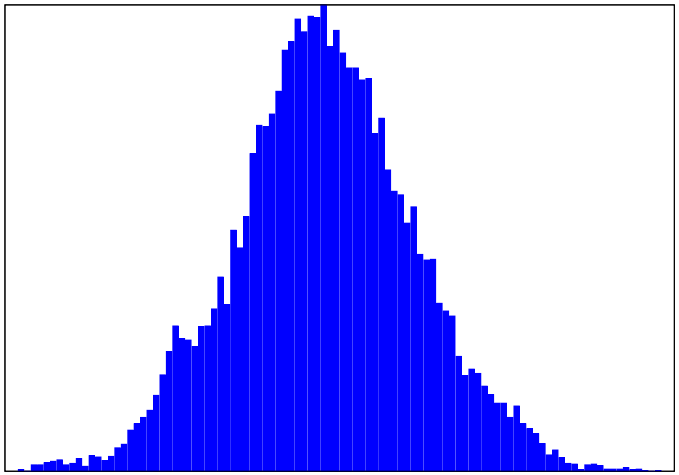
2nd iteration



Find a simple structure that describes the messages

- Histogram of $T_{i \to j}$'s

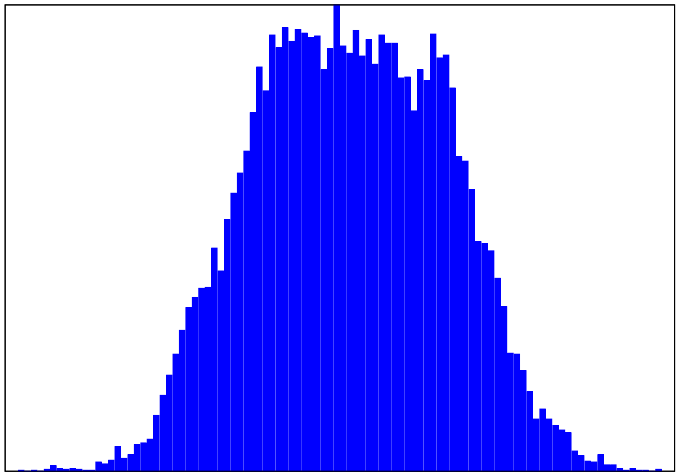
3rd iteration



Find a simple structure that describes the messages

- Histogram of $T_{i \to j}$'s

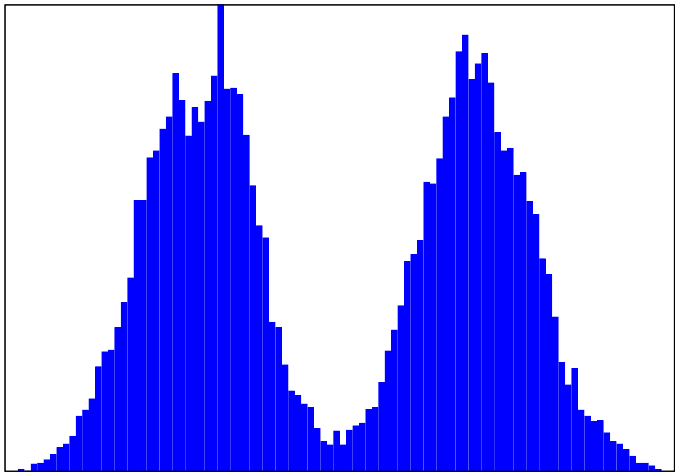
4th iteration



Find a simple structure that describes the messages

- Histogram of $T_{i \to j}$'s

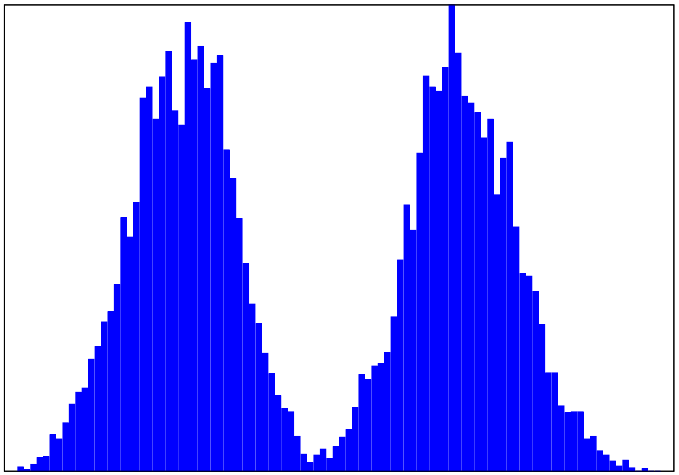
5th iteration



Find a simple structure that describes the messages

- Histogram of $T_{i \to j}$'s

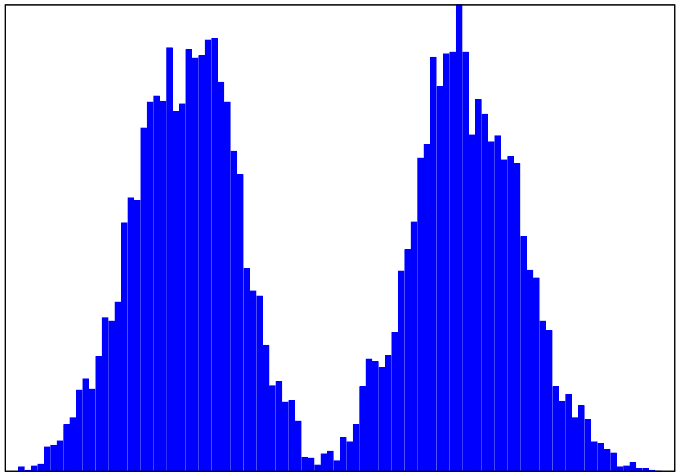
6th iteration



Find a simple structure that describes the messages

- Histogram of $T_{i \to j}$'s

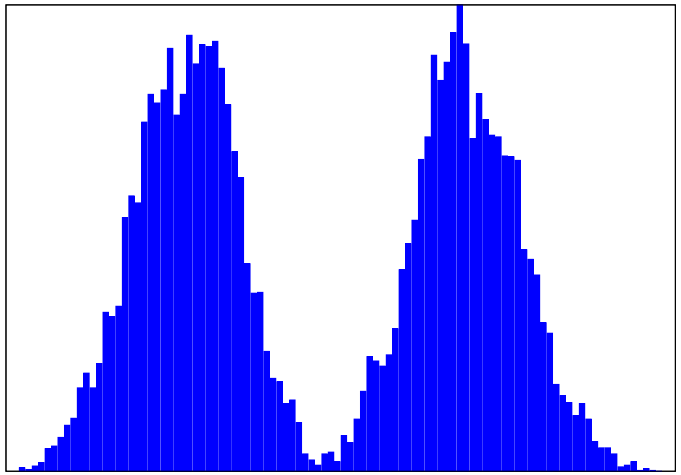
7th iteration



Find a simple structure that describes the messages

- Histogram of $T_{i \to j}$'s

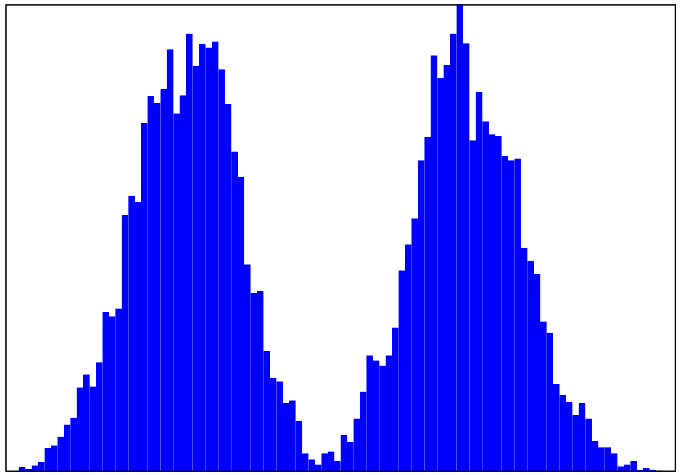
9th iteration



Find a simple structure that describes the messages

- Histogram of $T_{i \to j}$'s

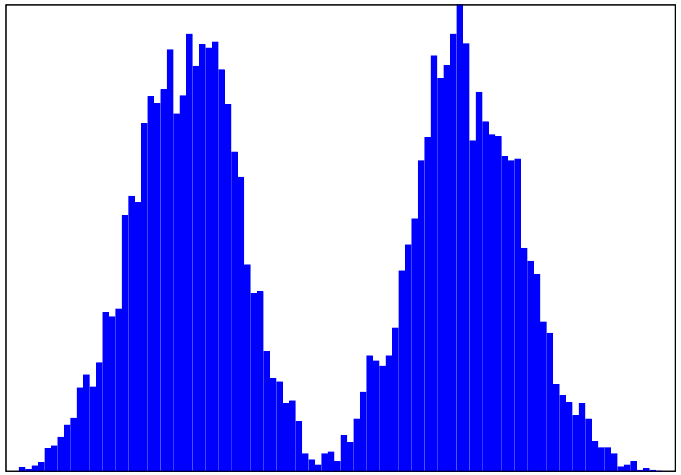
15th iteration



Find a simple structure that describes the messages

- Histogram of $T_{i \to j}$'s

20th iteration



Strategy: Find simple structure to describe the messages

- $\mathbf{T}^{(k)}$'s are sub-Gaussian
- Two parameters: mean m_k, scale factor σ_k^2
- Can be solved in a closed form

Sub-Gaussian random variables

- $\mathbf{X}$ is sub-Gaussian with mean m and scale factor σ^2 if

$$\mathbb{E}\left[e^{\lambda \mathbf{X}}\right] \;\leq\; e^{m\lambda + \frac{1}{2}\sigma^2\lambda^2}$$

- Applying Chernoff's inequality

$$\mathbb{P}(\mathbf{X} \leq m - a\sigma) \;\leq\; \mathbb{E}[e^{\mathbf{X}\lambda - m\lambda + a\sigma\lambda}] \;\leq\; e^{\frac{1}{2}\sigma^2\lambda^2 + a\sigma\lambda} \;\leq\; e^{-\frac{1}{2}a^2}$$

Sub-Gaussian random variables

- $\mathbf{X}$ is sub-Gaussian with mean m and scale factor σ^2 if

$$\mathbb{E}\left[e^{\lambda \mathbf{X}}\right] \;\leq\; e^{m\lambda + \frac{1}{2}\sigma^2\lambda^2}$$

- Applying Chernoff's inequality

$$\mathbb{P}(\mathbf{X} \leq m - a\sigma) \;\leq\; \mathbb{E}[e^{\mathbf{X}\lambda - m\lambda + a\sigma\lambda}] \;\leq\; e^{\frac{1}{2}\sigma^2\lambda^2 + a\sigma\lambda} \;\leq\; e^{-\frac{1}{2}a^2}$$

- $\widehat{\mathbf{T}}^{(k)}$ is sub-Gaussian with mean m_k and scale factor σ_k^2

$$\mathbb{P}(\widehat{\mathbf{T}}^{(k)} \leq 0) \;\leq\; e^{-\frac{m_k^2}{2\sigma_k^2}}$$

Sub-Gaussian random variables

- $\mathbf{X}$ is sub-Gaussian with mean m and scale factor σ^2 if

$$\mathbb{E}\left[e^{\lambda \mathbf{X}}\right] \;\leq\; e^{m\lambda + \frac{1}{2}\sigma^2\lambda^2}$$

- Applying Chernoff's inequality

$$\mathbb{P}(\mathbf{X} \leq m - a\sigma) \;\leq\; \mathbb{E}[e^{\mathbf{X}\lambda - m\lambda + a\sigma\lambda}] \;\leq\; e^{\frac{1}{2}\sigma^2\lambda^2 + a\sigma\lambda} \;\leq\; e^{-\frac{1}{2}a^2}$$

- $\widehat{\mathbf{T}}^{(k)}$ is sub-Gaussian with mean m_k and scale factor σ_k^2

$$\mathbb{P}(\widehat{\mathbf{T}}^{(k)} \leq 0) \;\leq\; e^{-\frac{m_k^2}{2\sigma_k^2}}$$

Density evolution on real-values messages
can be solved by establishing sub-Gaussianity

Experiments with real crowdsourcing tasks

- Learning similarities using Crowdsourcing
  - recommendations
  - searching

You wanted to get the tie on the top, but it was not available. Which one would be a better substitute?

- ☐ the tie the left
- ☑ the tie on the right

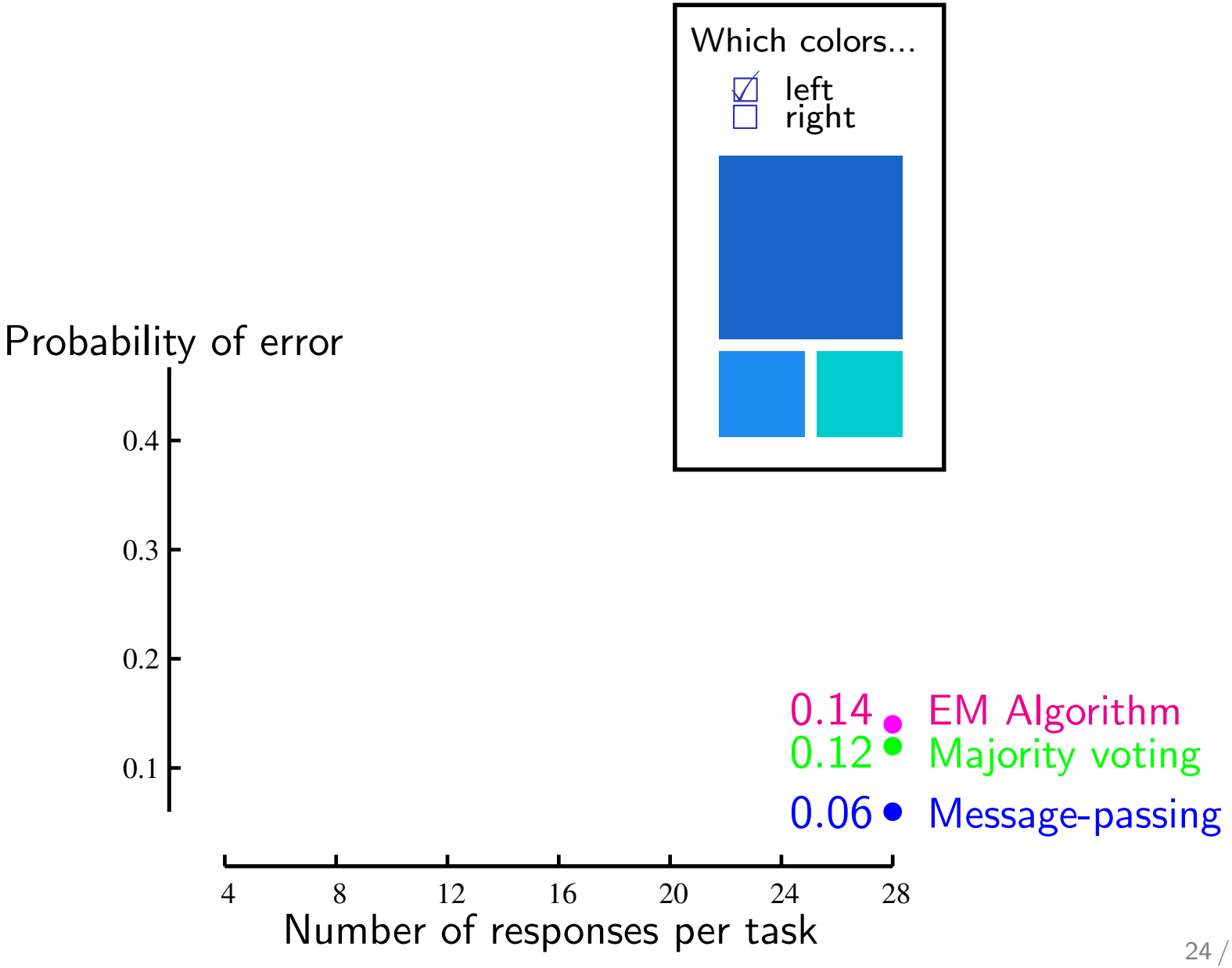
Experiments with real crowdsourcing tasks

- Crowdsourcing can handle much more complex tasks

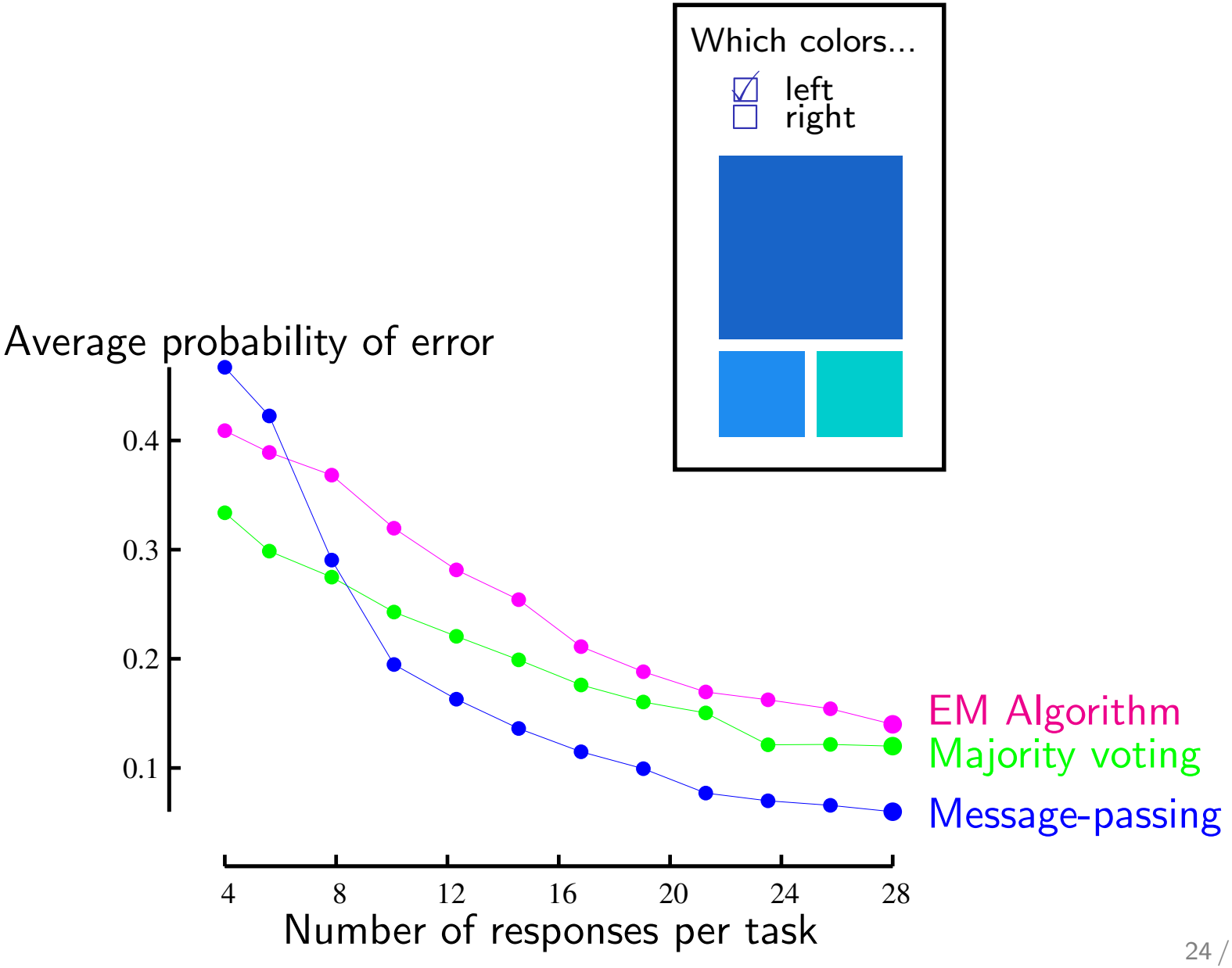
Which tie matches my shirt better?

- ☐ the tie on the left
- ☑ the tie on the right

Message-passing algorithm improves over existing algorithms

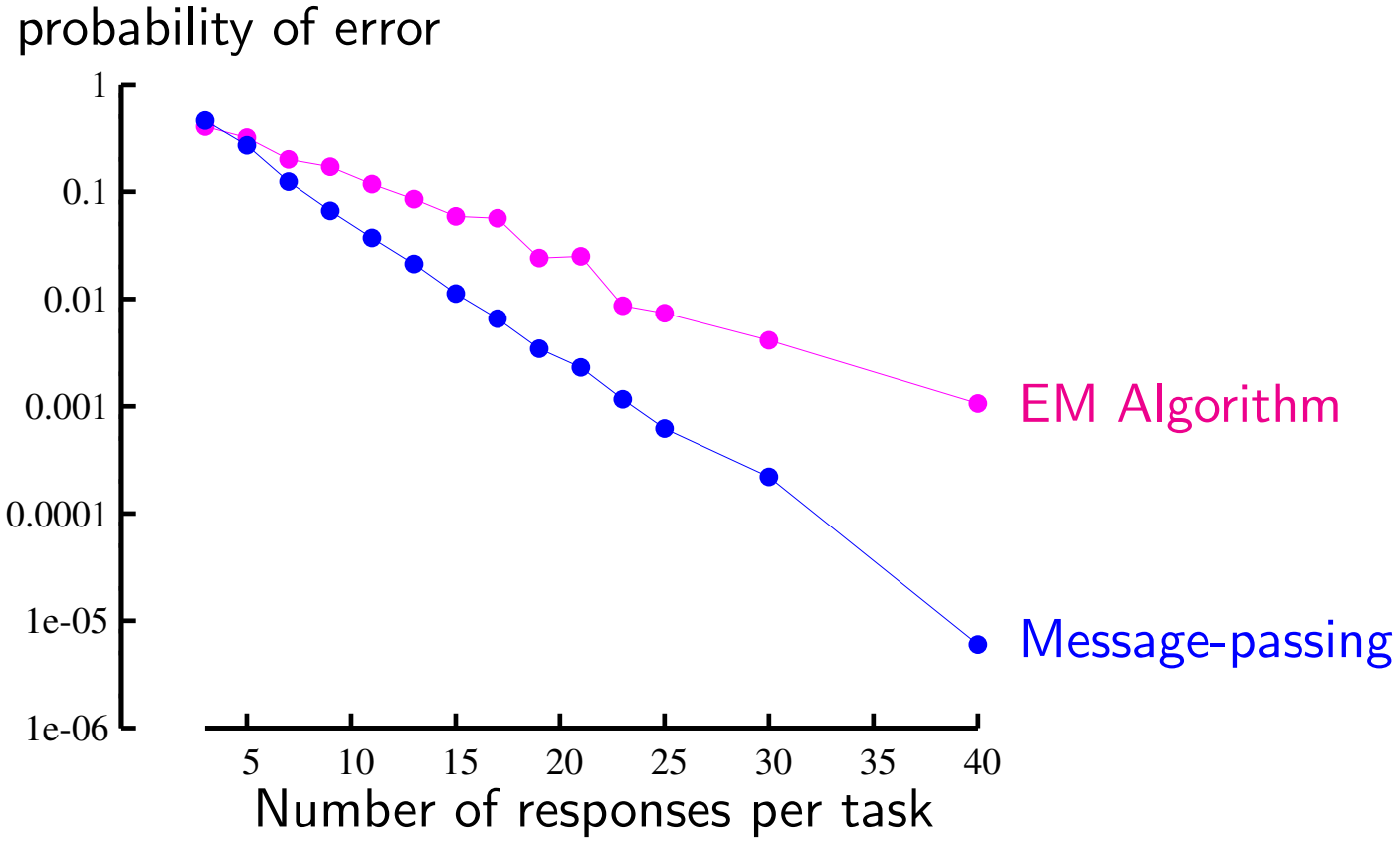


Message-passing algorithm improves over existing algorithms



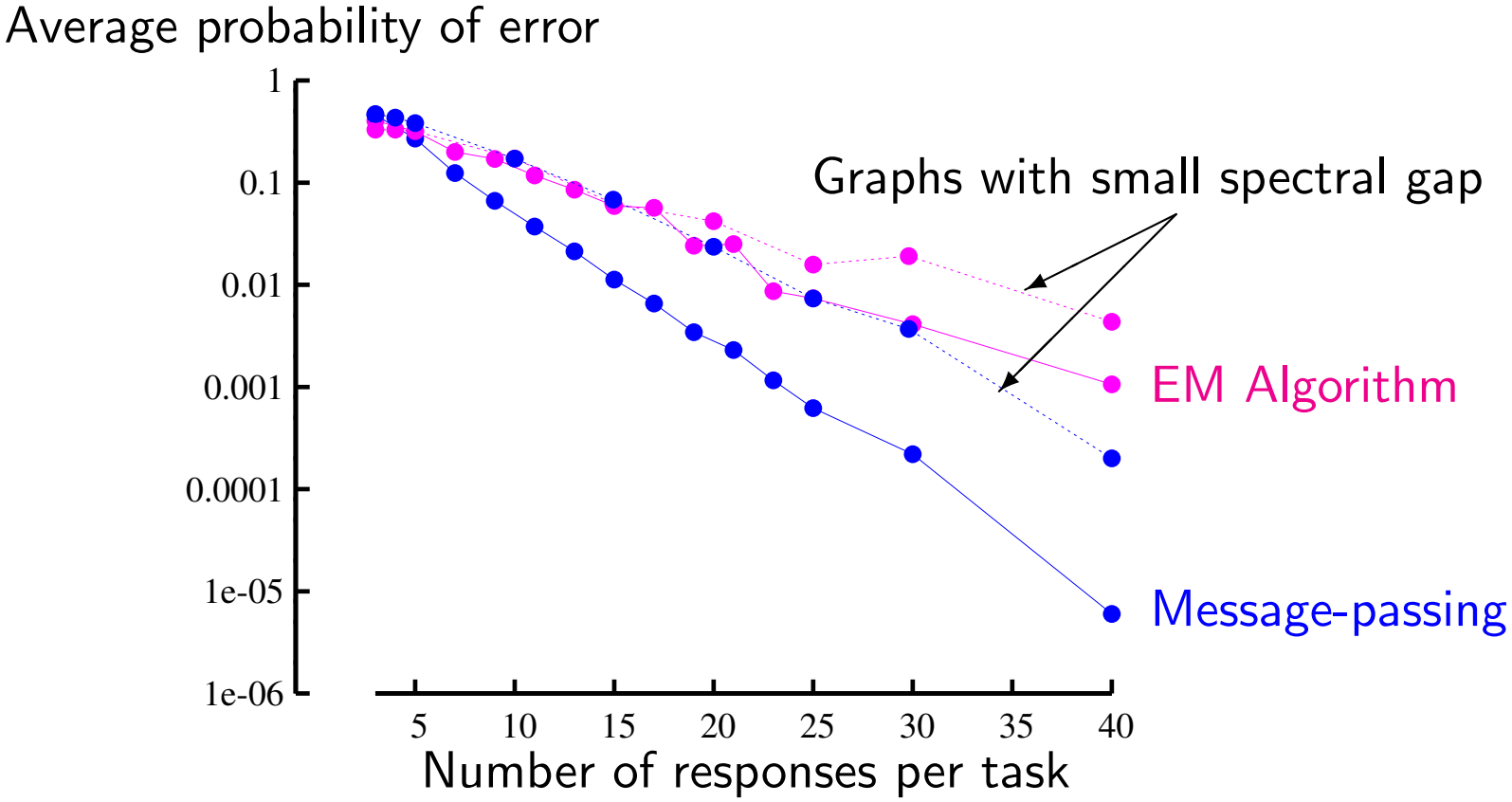
Gain due to random task assignment is significant

- How does the graph change the performance?



Gain due to random task assignment is significant

- How does the graph change the performance?

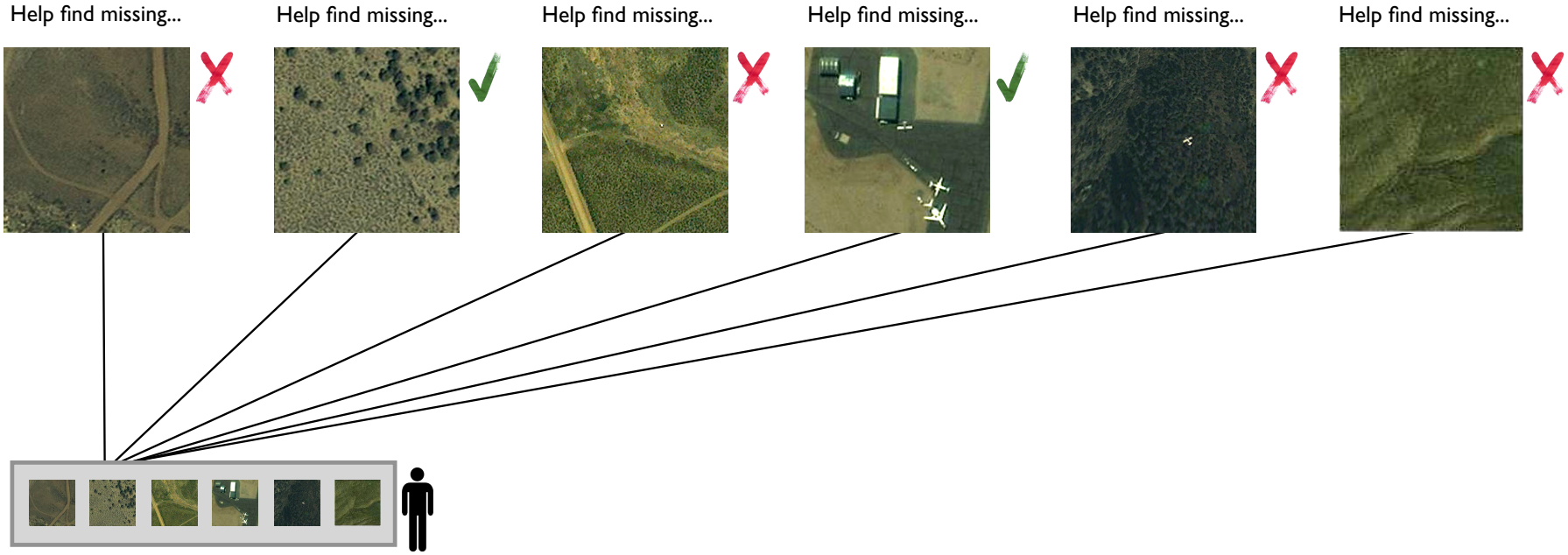


- Deterministic graphs with block-diagonal adjacency matrices
- Random graphs are good guidelines
  - Possible to generate deterministic graphs that are locally tree-like and good expanders

Under adaptive/sequential scenario

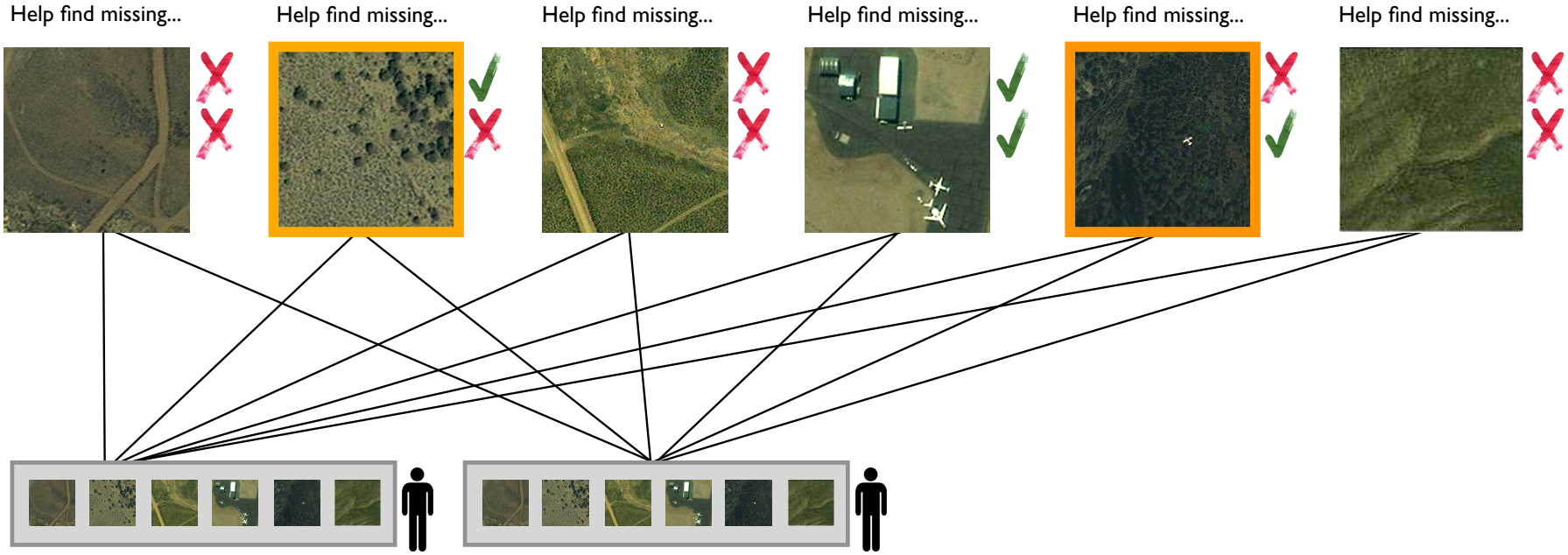
- Create batches adaptively
- Cannot explore/exploit reliable workers

Under adaptive/sequential scenario



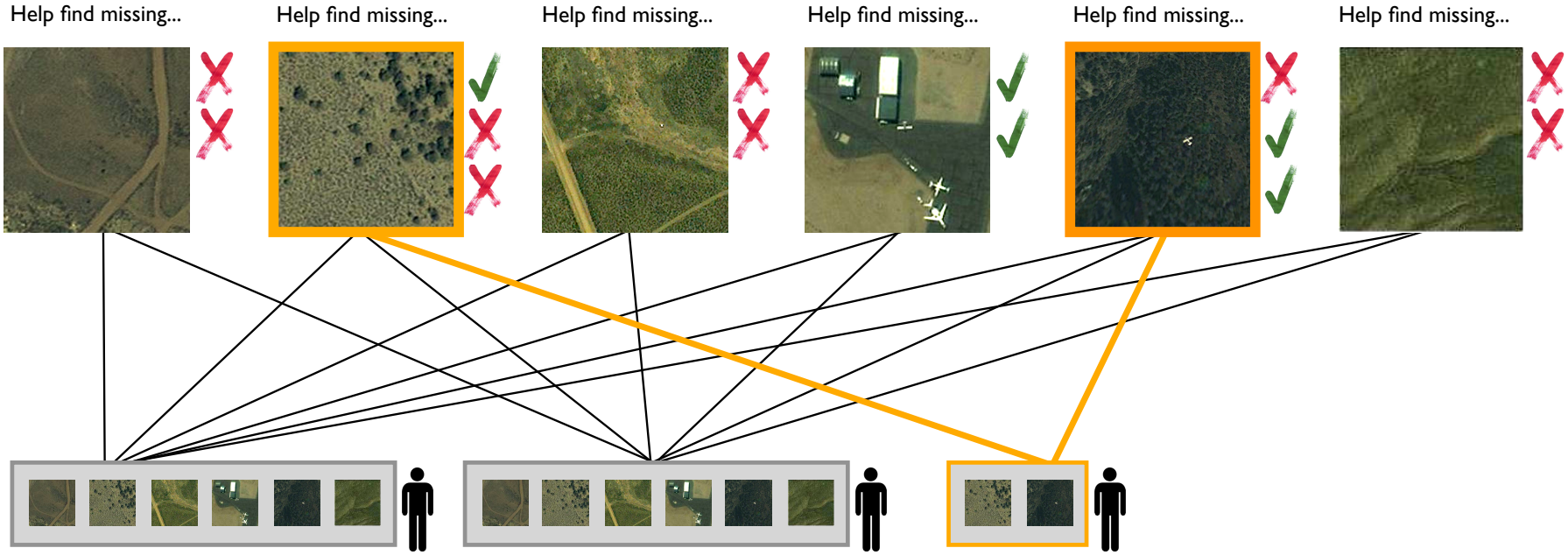
- Create batches adaptively
- Cannot explore/exploit reliable workers

Under adaptive/sequential scenario



- Create batches adaptively
- Cannot explore/exploit reliable workers

Under adaptive/sequential scenario



- Create batches adaptively
- Cannot explore/exploit reliable workers

Under adaptive/sequential scenario, our approach is still asymptotically order-optimal

Theorem. [Karger, O., Shah '11]

Let Δ be the budget necessary to achieve a target accuracy $\mathrm{P}_{\mathrm{error}} \leq \epsilon$. Then, for any Alg that can adaptively assign tasks,

$$\inf_{\mathrm{Alg}} \sup_{\{t_i\},\{p_j\}\in\mathcal{F}(q)} \mathbb{E}[\Delta] = \Omega\Big(\frac{1}{q}\log\Big(\frac{1}{\epsilon}\Big)\Big)$$

- Still Necessary to have budget $\geq C\,\frac{1}{q}\log(\frac{1}{\epsilon})$
- No significant gain in using an adaptive task allocation

When workers are biased

- Binary tasks: $t_i \in \{+1, -1\}$
- Worker reliability: $p_j^+, p_j^- \in [0, 1]$

If $t_i = +1$, then

$$A_{ij} = \begin{cases} t_i & \text{w.p. } p_j^+ \\ -t_i & \text{w.p. } 1 - p_j^+ \end{cases}$$

If $t_i = -1$, then

$$A_{ij} = \begin{cases} t_i & \text{w.p. } p_j^- \\ -t_i & \text{w.p. } 1 - p_j^- \end{cases}$$

When workers are biased

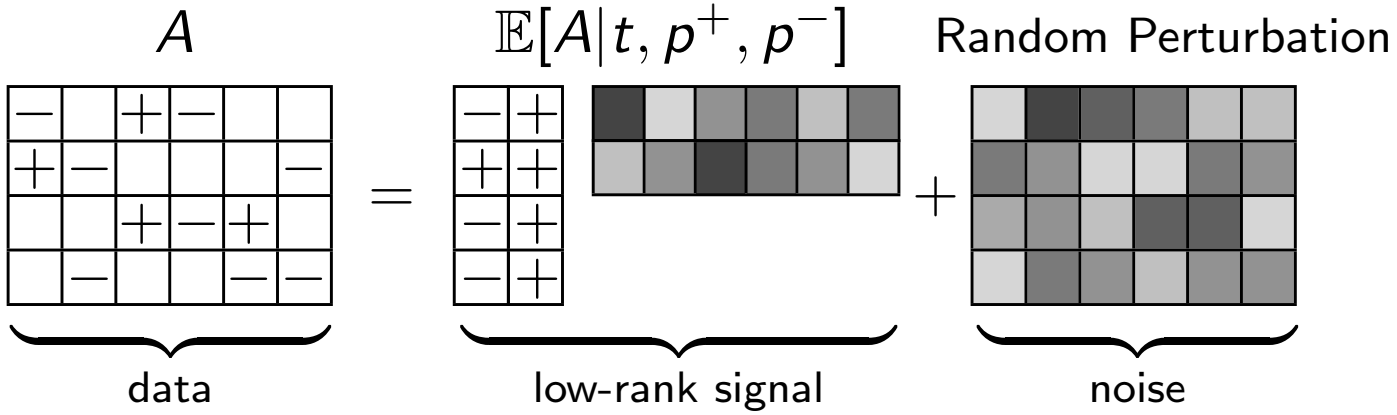
- Binary tasks: $t_i \in \{+1, -1\}$
- Worker reliability: $p_j^+, p_j^- \in [0, 1]$

If $t_i = +1$, then

$$A_{ij} = \begin{cases} t_i & \text{w.p. } p_j^+ \\ -t_i & \text{w.p. } 1 - p_j^+ \end{cases}$$

If $t_i = -1$, then

$$A_{ij} = \begin{cases} t_i & \text{w.p. } p_j^- \\ -t_i & \text{w.p. } 1 - p_j^- \end{cases}$$



Conclusion

Goal Design reliable and cost-efficient crowdsourcing systems

- Contributions
  - First to address both task assignment and inference problem
  - Message-passing Power Iteration Algorithm
  - Establish sub-Gaussianity to prove sharp bounds

- "Budget-optimal Task Allocation for Reliable Crowdsourcing Systems" available at arXiv:1110.3564