



# Ongoing Threads in Crowdsourcing Research

1>threads<1>threads

## Complexity

suzukiAtelier; KimStoria; yuanAlmost; Yu2016b;  
Nebeling:2016:WCW:2858036.2858169;  
Hahn:2016:KAB:2858036.2858364; crowdForgeKittur

## Workers

turkopticon; storiesIraniSilberman; crowdcollab;  
takingAHITMcInnis; dynamo; uberAlgorithm

## Decomposition

sensitiveTasks; LykourentzouPersonalityMatters;  
Law:2016:CKC:2858036.2858144;  
Chang:2016:ACC:2858036.2858411;  
Newell:2016:OMA:2858036.2858490



# Ongoing Threads in Crowdsourcing Research

2>threads<2>

## Complexity

suzukiAtelier; KimStoria; yuanAlmost; Yu2016b;  
Nebeling:2016:WCW:2858036.2858169;  
Hahn:2016:KAB:2858036.2858364; crowdForgeKittur

## Workers

turkopticon; storiesIraniSilberman; crowdcollab;  
takingAHITMcInnis; dynamo; uberAlgorithm

## Decomposition

sensitiveTasks; LykourentzouPersonalityMatters;  
Law:2016:CKC:2858036.2858144;  
Chang:2016:ACC:2858036.2858411;  
Newell:2016:OMA:2858036.2858490

# Complexity



What do we mean when we say **complexity**?

# Complexity



What do we mean when we say **complexity**?

- Can crowds help you **revise** work?  
bernsteinSoylent; Kim:2014:CSI:2556288.2556986;  
Nebeling:2016:WCW:2858036.2858169

# Complexity



What do we mean when we say **complexity**?

- Can crowds help you **revise** work?  
bernsteinSoylent; Kim:2014:CSI:2556288.2556986;  
Nebeling:2016:WCW:2858036.2858169
- Can crowds **critique** designs?  
yuanAlmost; fuge2014analysis

# Complexity



What do we mean when we say **complexity**?

- Can crowds help you **revise** work?  
*bernsteinSoylent; Kim:2014:CSI:2556288.2556986;  
Nebeling:2016:WCW:2858036.2858169*
- Can crowds **critique** designs?  
*yuanAlmost; fuge2014analysis*
- Can crowds create artifacts **de novo**?  
*KimStoria; Kim2017; Hahn:2016:KAB:2858036.2858364;  
Lasecki:2014:LSR:2661334.2661352*

# Complexity



... In some cases. Sometimes.

- Can crowds help you **revise** work?  
`bernsteinSoylent; Kim:2014:CSI:2556288.2556986;`  
`Nebeling:2016:WCW:2858036.2858169`
- Can crowds **critique** designs?  
`yuanAlmost; fuge2014analysis`
- Can crowds create artifacts **de novo**?  
`KimStoria; Kim2017; Hahn:2016:KAB:2858036.2858364;`  
`Lasecki:2014:LSR:2661334.2661352`

# Complexity



... In some cases. Sometimes.

- Can crowds help you **revise** work?  
bernsteinSoylent; Kim:2014:CSI:2556288.2556986;  
Nebeling:2016:WCW:2858036.2858169
  - Can crowds **critique** designs?  
yuanAlmost; fuge2014analysis
  - Can crowds create artifacts **de novo**?  
KimStoria; Kim2017; Hahn:2016:KAB:2858036.2858364;  
Lasecki:2014:LSR:2661334.2661352
- ⇒      With variable success

# Complexity



... In some cases. Sometimes.

- Can crowds help you **revise** work?  
bernsteinSoylent; Kim:2014:CSI:2556288.2556986;  
Nebeling:2016:WCW:2858036.2858169  
⇒ With variable success
- Can crowds **critique** designs?  
yuanAlmost; fuge2014analysis  
⇒ With careful guidance
- Can crowds create artifacts **de novo**?  
KimStoria; Kim2017; Hahn:2016:KAB:2858036.2858364;  
Lasecki:2014:LSR:2661334.2661352

# Complexity



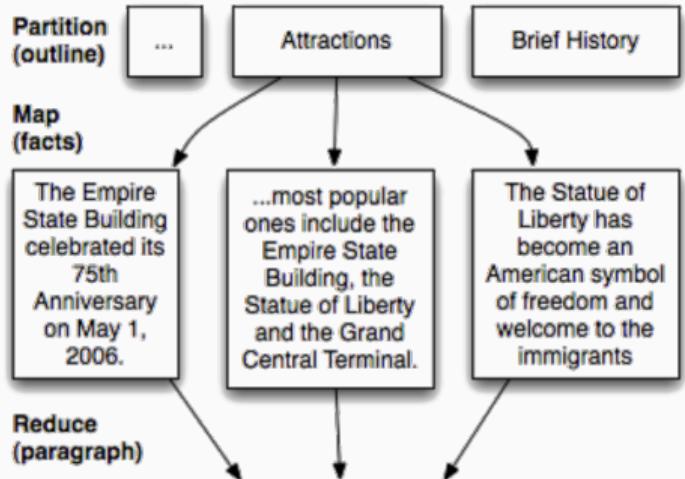
... In some cases. Sometimes.

- Can crowds help you **revise** work?  
*bernsteinSoylent; Kim:2014:CSI:2556288.2556986;  
Nebeling:2016:WCW:2858036.2858169* ⇒ With variable success
- Can crowds **critique** designs?  
*yuanAlmost; fuge2014analysis* ⇒ With careful guidance
- Can crowds create artifacts **de novo**?  
*KimStoria; Kim2017; Hahn:2016:KAB:2858036.2858364;  
Lasecki:2014:LSR:2661334.2661352* ⇒ Within narrow specifications

# What Does On-Demand Work Say?

Build complexity into the process

- Apply CS methods to people  
[crowdForgeKittur](#)

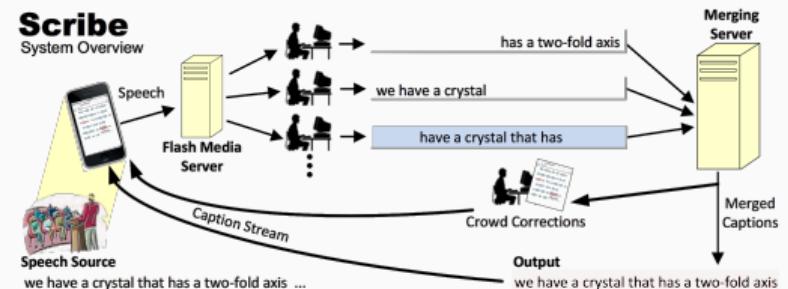


# What Does On-Demand Work Say?



Build complexity into the process

- Apply CS methods to people  
**crowdForgeKittur**
- Humans as computational units  
**Lasecki:2014:LSR:2661334.2661352**

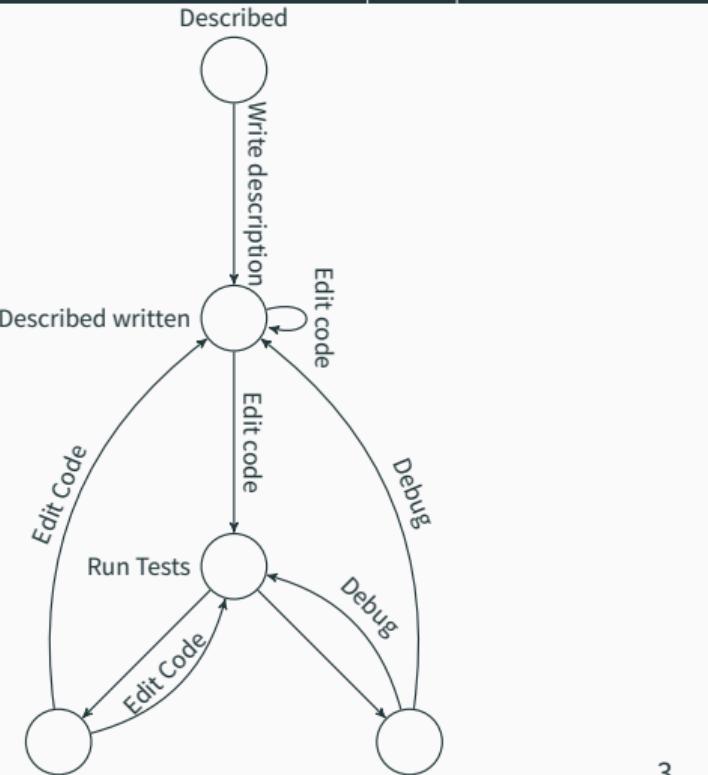


# What Does On-Demand Work Say?



Build complexity into the process

- Apply CS methods to people  
[crowdForgeKittur](#)
- Humans as computational units  
[Lasecki:2014:LSR:2661334.2661352](#)
- Crowdsourcing workflows as function state machines  
[latoza2014microtask](#)





## What Does Piecework Say?

What we'll find

- Building complexity *into* processes



## What Does Piecework Say?

What we'll find

- Building complexity *into* processes
- Incremental advances until managers *tracked* and *standardized* workers and work



## What Does Piecework Say?

What we'll find

- Building complexity *into* processes
- Incremental advances until managers *tracked* and *standardized* workers and work
- Challenges with *flexibility*



## What Does Piecework Say?

What we'll find

- Building complexity *into* processes
- Incremental advances until managers *tracked* and *standardized* workers and work
- Challenges with *flexibility*
- Insights into task *specialization*

# What Does Piecework Say?

George Airy. Astronomer. Crowd work requester.

grier2013computers



- Employed computers
- 13–20 years old
- no particularly strong background in mathematics
- A basic understanding of logarithms, algebra, etc...



# George Airy

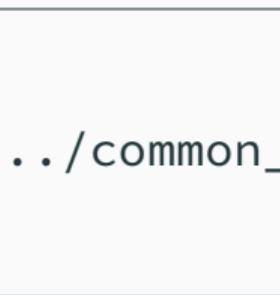
Airy built complexity into the process, assigning *human computers* to calculate astronomical movements.

No. of Swings.	Approximate Time (Astronomical Reckoning).	Number of Signals.	Mean of Times by SHELTON.	Mean of Times by EARNSHAW.	Interval by SHELTON.	Interval by EARNSHAW.	Rate EARNSHAW / SHELTON	Logarithm of EARNSHAW / SHELTON	Corrected Logarithm of Rate EARNSHAW / SHELTON
	Oct. h		h m s	h m s	h m s	h m s			
1....	1. 23	22	3 19 36.505	21 23 28.764	...4 0 23.100	4 0 38.722	1.0010831	0.00047012	
2....	2. 3	21	7 19 59.605	1 24 7.486	...3 58 21.652	3 58 37.400	1.0011011	0.00047793	
3....	2. 7	21	11 18 21.257	5 22 44.886	...4 45 27.829	4 45 46.421	1.0010855	0.00047117	0.00047387
4....	2. 11	29	16 3 49.086	10 8 31.307	...4 17 6.532	4 17 23.234	1.0010827	0.00046995	
5....	2. 16	17	20 20 55.618	14 25 54.541	...3 13 21.898	3 13 34.795	1.0011116	0.00048249	
6....	2. 19	25	23 34 17.516	17 39 29.336	...3 49 42.503	3 49 57.654	1.0010994	0.00047720	0.00047990
7....	2. 23	31	3 24 0.019	21 29 26.990	...3 55 2.071	3 55 17.433	1.0010893	0.00047282	
8....	3. 3	21	7 19 2.090	1 24 44.423	...4 2 41.510	4 2 57.445	1.0010944	0.00047503	
9....	3. 7	25	11 21 43.600	5 27 41.868	...4 31 5.786	4 31 23.591	1.0010947	0.00047516	0.00046316
10....	3. 11	22	15 52 49.386	9 59 5.459	...3 27 49.747	3 28 3.324	1.0010888	0.00047260	
11....	3. 15	24	19 20 39.133	13 27 8.783	...3 59 47.292	4 0 3.188	1.0011049	0.00047959	
12....	3. 19	24	23 20 26.425	17 27 11.971	...4 3 30.416	4 3 46.629	1.0010686	0.00046384	0.00047194

## Low Complexity



Farms



.../common\_figures/farm payment for results

**10.2307/2338394**

- Formalization of piecework:

**10.2307/2338394**

- Dynamic piece rates

## Low Complexity



- Distributed workers

Textiles



.../common

figures/pieceworkers.jpg  
common skills

## Low Complexity

- Strict management
- Formalizing work methods

Matchstick Girls

.. /common\_figure



# Low Complexity



Farms

Textiles

Matchstick Girls

..../common\_figures/farm-walkthroughs/pieceywalkthroughs/p

# Planes, Trains, and Automobiles



## Trains

[..../common\\_figures/photo/train\\_workers.jpg](#)

- “Efficiency experts” measured how long it would take to do various jobs

[doi:10.2307/1883615](#)

- These measurements would be used to assign pay rates for each specific task

[american1921problem](#)

# Planes, Trains, and Automobiles



## Automobiles

- Consolidating and training workers (*Fordism*)  
schoenberger1988fordism;  
tolliday1986between

.../common\_figures/photo/ford\_assembly\_line.jpg

- Measuring and evaluating workers by very carefully defined instructions (*Taylorism*) taylor1914principles

# Planes, Trains, and Automobiles

- Men drafted during World War II
- Factories turned to a new workforce who had neither conventional training nor experience
- **Specialized training and assignment**

Planes



.../common\_figures/photo/Rosie\_

# Planes, Trains, and Automobiles



Trains

Automobiles

Planes

.. /common\_figures /photo /common\_figures /photo /common\_figures /photo

## Comparisons



- Building complexity into the processes
- Challenges dealing with flexibility
  - *Building planes versus fixing trains*



## Implications for On-Demand Work

Has technology shifted on-demand work?

In some ways, yes

- Technology makes *some* complex tasks relatively trivial
- Measuring workers is easier than ever

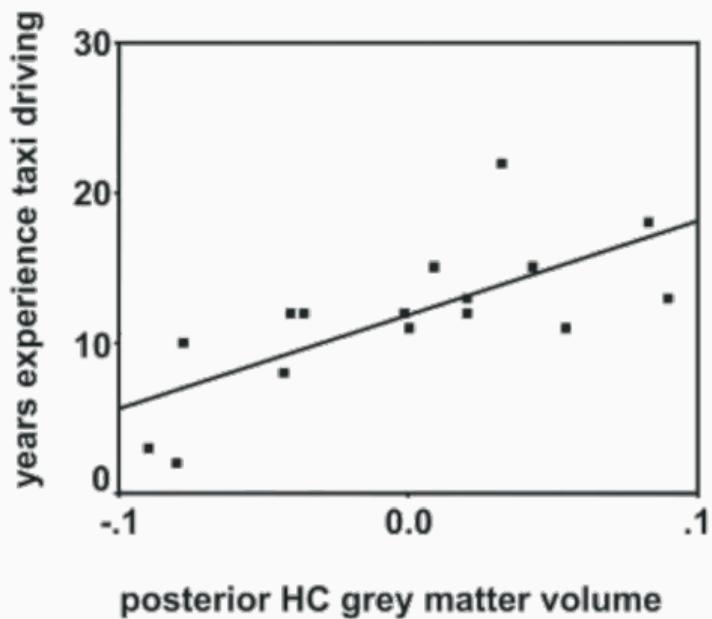
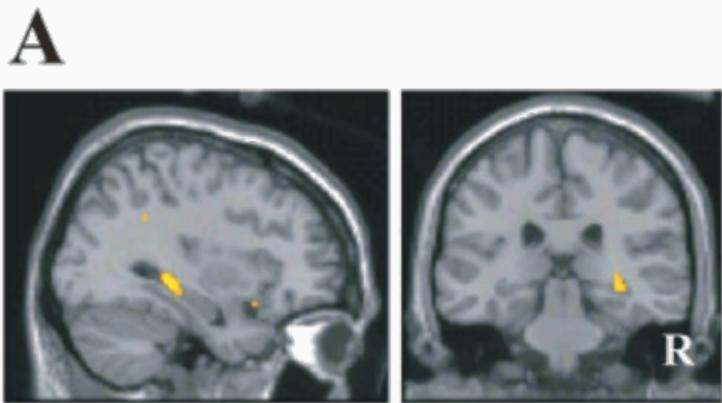
In other ways, no — we still don't have good end-to-end processes for arbitrarily complex work

*We can make a routine out of building planes, but not out of fixing trains*

# Enhanced Cognition



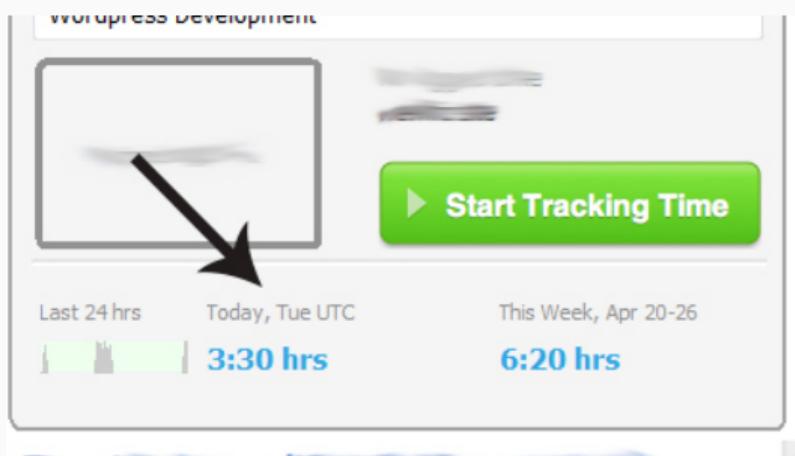
# Enhanced Cognition



# Enhanced Cognition



# Tracking Work and Workers



Upwork has turned to logging workers' keystrokes and taking screenshots automatically every 10 minutes

## Takeaways



- We make stronger assumptions about workers' abilities thanks to technology
- Evaluation remains difficult, but we're trying to find stopgap solutions through decomposition
- We're still not solving the problems of inherently subjectively judged work