Identifying the Complexity Limits of On–Demand Work

A key question to the future of on–demand work is *what* precisely will become part of this economy. Paid crowdsourcing began with simple microtasks on platforms such as Amazon Mechanical Turk, but microtasks are only helpful if they build up to a larger whole. So, our first question: how complex can the work outcomes from on–demand work be?

Crowd work's perspective

Crowdsourcing research has spent the better part of a decade proving the viability of crowdsourcing in complex work. Unless on–demand work as a whole can demonstrate viability for meaningfully complex tasks, the argument runs, it will be incapable of ensuring a pro–social outcome for work and workers [26]. Kittur et al. first opened the question of whether crowdsourcing could be used for goals that are not simple parallel tasks [25]. Their work demonstrated proof–of–concept crowdsourcing of a simple encyclopedia article and news summary — tasks which could be verified or repeated with reasonable expectations of similar outcomes. Seeking to raise the complexity ceiling [39], researchers have since created additional proof–of–concept applications and techniques, including conversational assistants [29], medical data interpreters [29], and idea generation work–flows [56, 54, 55], to name a few examples.

To achieve complex work, this body of research has often applied ideas from Computer Science to design new crowdsourcing work–flows. Beginning with a goal that has presented significant challenges for computers, the researcher leverages an insight from Computer Science (for example, MapReduce [25] or sequence alignment algorithms [28]) and arranges humans as computational black boxes within those approaches. This approach has proven a compelling one because it leverages the in–built advantages of scale, automation, and programmability that software affords.

It is now clear that this computational workflow approach works with focused complex tasks, but the broader wicked problems largely remain unsolved [42]. As a first example, idea generation shows promise [56, 54, 55], but there is as yet no general crowdsourced solution for the broader goal of invention and innovation [14]. Second, focused writing tasks are now feasible [24, 5, 40, 48, 1], but there is no general solution to create a cross–domain, high–quality crowd–powered author. Third, data analysis tasks such as clustering [10], categorization [4], and outlining [34] are possible, but there is no general solution for sense–making. It is not yet clear what insights would be required to enable crowdsourced solutions for these broader wicked problems.

Restricting attention to non–expert, microtask workers proved limiting. So, Retelny introduced the idea of crowdsourcing with online paid *experts* from platforms such as Upwork. Expert crowdsourcing enables access to a much broader set of workers, for example designers and programmers. The same ideas can then be applied to expert "macro–tasks" [9, 18], enabling the crowdsourcing of goals such as user–centered design [41], programming [30, 13, 8], and mentorship [47]. However, there remains the open question of how complex the work outcomes from expert crowds can be.

Piecework's perspective

Piecework's body of research most squarely addresses complexity in two of the cases we looked at earlier: Airy's human computers and among industrial workers, especially during World War II. The majority of piecework's perspective, on this topic, will come from these areas.

Airy's work opened the door to greater task complexity by decomposition. Airy's computers had relatively limited education in mathematics, but they were nevertheless collectively able to accomplish complex goals by approaching distinct, subdivided, quickly verifiable tasks in chunks [17].

It seemed, too, that only tasks that could be measured and priced could be completed via piecework — a conclusion which would fundamentally limit the potential complexity of piecework. Earlier we discussed Graves's and later Brown's analysis of railway workers; one might conclude from Graves's observations in particular that complex, creative work — which is inherently heterogeneous and difficult to routinize — would be unsuitable for piecework [16]. Brown's description of efficiency experts would corroborate this; "efficiency experts" can effectively gauge how long known tasks should take, but would find themselves overwhelmed if they attempted to assess creative work like design, which can take an arbitrary number of iterations before proceeding to a subsequent step.

The next innovation in piecework's drive toward complexity came on the assembly line, specifically during the war effort. At a time when manufacturing firms needed more able–bodied workers, but all of the conventionally trained metalworkers had joined the armed forces, factories turned to women [22, 11]. This need to train workers rapidly to do very narrowly defined tasks resulted in a *recomposition* of work, wherein workers were only prepared for the tasks they would do; riveters would only be trained on how to do rivet and to pass the output to the next person on the assembly line, who could evaluate and send the work back or proceed with the next step.

Only tasks that could be measured and priced could be completed via piecework — a fact which fundamentally limited the potential complexity of piecework.

and difficult to routinize — would be unsuitable for piece-work [16]. Brown's description of efficiency experts would corroborate this; "efficiency experts" can effectively gauge how long known tasks should take, but would find themselves overwhelmed if they attempted to assess creative work like design, which can take an arbitrary number of iterations before proceeding to a subsequent step.

Piecework was limited to tasks that could be quickly and accurately evaluated. For example, the roles required to facilitate piecework in the early 20th century included "piecework clerks, inspectors, and 'experts'" [16]. Hart argue that evaluation limited piecework's complexity: at some point, evaluating multidimensional work for quality (rather than for quantity) becomes infeasible. In his words, "if the quality of the output is more difficult to measure than the quantity [...] then a piecework system is likely to encourage an over–emphasis on quantity ... and an under–emphasis on quality" [20].

Piecework researchers were also finding that certain types of *organizations*, as well as certain types of *work*, were more amenable to piecework regimes. Brown wrote that piecework was "less likely in jobs with a variety of duties than in jobs with a narrow set of routinized duties" [7]; Agell hypothesized this as an effect of market forces — suggesting that "pay schemes based on ... performance may induce workers to neglect tasks that are less easy to measure" [2]. Airy seemed to recognize this, decomposing complex mathematical problems into easily verified components, but applying this model to tasks like railway engineering may have been prohibitively difficult.

But it was more than the inherent qualities of work and the organization of the firm that mattered; management practices themselves had to be appropriately geared to working with pieceworkers. Competent foremen were integral both for allocating resources and for liaising between workers and upper management [53, 6].

]

Comparing the phenomena
The research on piecework tells us that we should expect piecework to thrive in industries where the nature of the work is limited in complexity [7]. Given the flourishing of on–demand labor platforms such as Uber, AMT, and others, we ask ourselves what — if anything — has changed. We argue that the internet has trivialized the costs and challenges of the earlier limiting factors because technology makes it easier 1) for workers to do complex work without training, 2) to manage workers in doing complex work, and 3) to create the infrastructure necessary to manage the workers.

Technology increases non–experts' levels of expertise by giving access to information that would otherwise be unavailable. For example, taxi drivers in London endure rigorous training to pass a test known as "The Knowledge" — a demonstration of the driver's comprehensive familiarity with the city's roads. This test is so challenging that veteran drivers develop significantly larger the regions of the brain associated with spatial functions such as navigation [36, 35, 45, 46, 52, 51]. In contrast, with on–demand platforms such as Uber, services such as Google Maps & Waze make it possible for people en-

tirely unfamiliar with a city to operate professionally [44, 21]. Other examples include search engines enabling information retrieval, and word processors enabling spelling and grammar checking. By augmenting the human intellect [12], computing has shifted the complexity of work that is possible without training.

Algorithms have automated some tasks that previously fell to management. Computational systems hire workers [33, 49], as well as direct their activities [31], and act as "piecework clerks" [16] to inspect, modify and combine work [23, 38]. In many cases, the intermediary function has been removed as well, leading workers to need to directly email requesters for clarification and feedback [37]. These algorithms, however, are less able than human managers to manage contingencies that were not programmed into them.

Finally, the organizational limit on infrastructure creation is somewhat lessened. Writing web scripts takes fewer people and fewer hours than creating physical equipment for piece-work. Little et al.'s vision was that any user with basic programming skills could tap into on–demand human intelligence. As better toolkits lower this threshold [39] and computational thinking diffuses, a broader population will be able to use on–demand work.

Implications for crowd work
Technology's ability to support human cognition will enable stronger assumptions about workers' abilities, increasing the complexity of on–demand work outcomes. Just as the shift to expert crowdsourcing increased complexity, so too will workers with better tools increase the set of tasks possible. Beyond this, further improvements would most likely come from replicating the success of narrowly–slicing education for expert work as Hart and Roberts and Grier described in their piecework examples of human computation [17] and drastically reformulating macro–tasks given the constraints of piecework [19]. To some extent, an argument can be made that MOOCs and other online education resources provide crowd workers with the resources that they need, but it remains to be seen whether that work will be appropriately valued, let alone properly interpreted by task solicitors [3]. If we can overcome this obstacle, we might be able to empower more of these workers to do complex work such as engineering and metalworking, rather than doom them to "uneducated" match girl reputations [43]. However, many such experts are already available on platforms such as Upwork, so training may not directly increase the complexity accessible to on–demand work unless it makes common expertise more broadly available.

Will the shift from human managers to Turing–complete algorithms raise the complexity ceiling? By the Turing test, the algorithms would be at best indistinguishable from human piecework clerks and foremen. So in terms of enabling coordination, algorithmic management is unlikely to directly raise the ceiling beyond what piecework could achieve. However, as a resource constraint, algorithms are a fixed cost and not a per–person cost like human managers. So in terms of accessibility, algorithms will allow a broader class of organizations and individuals to benefit from soliciting on–demand work. This

shift may enable complex goals that were not cost–effective before to become feasible. However, because algorithms remain far from replicating all of the foreman's responsibilities, most likely is a middle ground in which on–demand work re–introduces the human element to management in a more targeted way (e.g., [18, 27, 50]). This move will require resolving the tension between workers and perilously antagonistic managers, as Boal and Pencavel suggest, to break a toxic cycle of mistrustful requesters [15].

Finally, the cost of creating piecework infrastructure has dropped. Expensive manufacturing equipment has been largely replaced by computer code [32]. As with lowered costs of management, lowered infrastructure costs will make on–demand work accessible to a broader set of people and organizations. This in and of itself does not raise the complexity ceiling, but by broadening the potential market for on–demand work, it may enable a new set of goals and needs take part.