Analytics for Refugee Resettlement

Narges Ahani

A Dissertation

submitted to the faculty of the

WORCESTER POLYTECHNIC INSTITUTE

in partial fulfillment of the requirements for the

Degree of Doctor of Philosophy

in

Data Science

April 2022

Approved:

_____

Professor Andrew C. Trapp (WPI, Advisor)

_____

Professor Randy Paffenroth (WPI)

_____

Professor Osman Özaltın (North Carolina State University)

_____

Professor Alexander Teytelboym (University of Oxford)

Abstract

Every year tens of thousands of refugees are resettled to dozens of host countries. While there is growing empirical evidence that the initial placement of refugee families profoundly affects their lifetime outcomes such as employment, there have been few attempts to optimize resettlement decisions. This dissertation is about the use of *analytics* in refugee resettlement. We leverage machine learning, matching theory, integer optimization, stochastic programming, risk modeling, and interactive visualization to improve the refugee resettlement decision-making process and benefit involved stakeholders including refugees, communities and resettlement agencies.

First, recognizing that specific synergies exist between refugee characteristics and resettlement communities that affect refugee employment outcomes, we use machine learning to train predictive models on past refugee placement and outcome data to estimate employment likelihood of refugees in communities. These estimated values are then used as refugee-community match quality scores in an integer optimization model for optimal matching of to-be-arriving refugees into the network of communities. We implemented our analytical approaches into an innovative, interactive refugee resettlement decision support software, *Annie*™ Moore, that assists HIAS, one of the nine U.S. resettlement agencies, with matching refugees to their initial placements by providing optimized, data-informed recommendations. Our software suggests optimal placements while giving substantial autonomy to the resettlement staff to fine-tune recommended matches through interactive visualization, thereby streamlining their resettlement operations. Backtesting indicates that *Annie*™ can improve short-run employment outcomes by 22%–38%.

Second, we consider the *dynamic* nature of refugee resettlement. While refugees arrive weekly, capacities are assigned to communities on an annual basis. By only allowing resettlement staff to manually set *weekly* capacities, this may result in consuming annual capacities in an overly greedy or conservative manner. In other words, allocating the weekly batches of arriving refugees with arbitrary weekly community capacity settings leads to sub-optimal total annual employment, as each batch of arrivals is allocated by separately maximizing the expected employment of this batch and without considering future arrivals. While the optimized value for total annual expected employment can only be realized when perfect information is available for all future arrivals, to better approximate this hindsight optimal employment we introduce a dynamic allocation system based on two-stage stochastic programming to improve employment outcomes. Our algorithmic approach places refugees to communities using not only employment probabilities, but also estimates of the value of the remaining slots of capacity for each community, leveraging this critical information on whether each slot is more useful for placing the current refugee or a yet-unknown refugee arriving later in the year. This algorithm is able to achieve over 98% of the hindsight-optimal employment compared to less than 90% for existing myopic approaches. We recently incorporated our dynamic placement algorithm into *Annie*™.

Third, we extend our earlier optimization model to account for *risk*. We recognize that inherent error exists with respect to the estimation of employment probabilities. This results in uncertainty with respect to expected optimized outcomes for refugees, that is, employment likelihoods for optimized refugee-community placements. Directly related to this uncertainty, we introduce the concept of risk in refugee resettlement optimization. Although numerous studies exist on risk in the context of optimization, the related literature largely interprets

risk as expressed by uncertainty around total expected objective function value. Considering that the expected outcome of each refugee family—the employment likelihood at its optimal placement—is just as important as the total expected employment from maximizing all refugee-community placements, we provide an alternative *family-level* definition of risk that properly accounts for vulnerability of refugees and is useful in the context of refugee resettlement. We seek to mitigate this alternative definition of risk from an optimization point of view. Our modeling approach explicitly incorporates family-level risk into the formulation by accounting for both the total expected outcomes as well as risk related to placement outcome uncertainty for refugee families. To hedge against this risk, we weigh the trade-off in lower expected outcomes associated with less risk, generating optimal solutions that satisfy risk-averse decision makers. Optimizing this trade-off presents significant modeling and computational challenges. Multiple optimization models are proposed and analyzed and specific measures are developed to quantify the change in risk. We discuss on the relative benefits and trade-offs of these models and provide experimental results to illustrate their performance. Our results show that risk-averse optimization models can alleviate much of the risk while retaining much of the total expected employment.

Acknowledgements

I am incredibly grateful to many for helping make this dissertation successful.

First and foremost, I would like to express my sincere gratitude to my advisor, Professor Andrew C. Trapp, who gave me invaluable advice and unwavering support in both academics and life. I am incredibly thankful for his patience during my PhD study, his understanding, encouragement and belief in me. I would like to thank him for sharing his immense knowledge and plentiful experience with me to help me in my academic research and daily life.

I would like to offer my special thanks to Professor Alexander Teytelboym who inspired me and directly influenced my academic journey.

I would like to extend my sincere thanks to Professor Alessandro Martinello, Paul Gölz, Professor Ariel D. Procaccia, and Professor Osman Özaltın, without their knowledge and assistance, this dissertation thesis would have never been accomplished.

I am deeply grateful to Professor Tommy Andersson and Michael Ginzberg for providing the funding opportunity to undertake my studies at Worcester Polytechnic Institute.

I am extremely grateful to Mike Mitchell, Alicia Wrenn and Karen Monken and other members of HIAS for their support and assistance.

I would like to thank my dissertation committee member Professor Randy Paffenroth for his time and advice.

I would like to acknowledge the financial support of the National Science Foundation (Operations Engineering) grant CMMI-1825348, the financial support of the Jan Wallander and Tom Hedelius Foundation (Research Grant P2016-0126:1) and the Ragnar Söderberg Foundation (E8/13), as well as that provided by the Business School at Worcester Polytechnic Institute.

Most importantly, I would like to extend my special thanks to my family and friends for their love and support and to everyone, who directly or indirectly, have lent their hand in this chapter of my life.

Contents

Introduction

At the end of 2020, there were 82.4 million forcibly displaced people worldwide due to persecution, conflict, violence or human rights violations [1]. Around 20.7 million are refugees who have crossed international borders and mostly live within cities or established camps in the countries of asylum which are mostly countries neighbouring their country of origin [2]. Many studies investigated the scope of refugee camp management [3]. These works mainly study the different aspect of managing refugee camps from decision making regarding the location for establishing refugee camps to daily operational decisions in the administrative level of camp life. As stated in [4], living in camp is considered as a temporary condition for refugees. While a lasting solution for displaced people to rebuild their lives is to return to their home countries under safe conditions. But because conflicts continue in many areas, returning home safely is not a viable option for many refugees. Another solution is local integration which refers to the settlement of refugees with full legal rights in the country to which they have fled (country of asylum). However, overburdened asylum systems limit possibilities of local integration too. Thus *resettlement* continues to be a critical solution, that is to resettle some of the world's most at risk or vulnerable refugees in third countries, ensuring protection and providing solutions for those who have specific or urgent protection needs. Unlike living in camp, resettling refugees in a third country can provide opportunities like employment and contributing economically to host communities.

In 2019, of the nearly 26 million refugees worldwide, the United Nation High Commission for Refugees (UNHCR) projected 1.44 million refugees to be in need of resettlement. In the same year, 81,671 were submitted to be considered for resettlement to 29 countries, of which 63,726 refugees actually departed for resettlement. This represents ***less than 4.5%*** of the global resettlement needs in that year, meaning only a small fraction of those at significant risk found a safe and lasting solution [5].

While *analytics*, the scientific transformation of data into insight for better decision making, has predominantly seen application in, and been driven by, competitive settings of private sectors and companies, these tools are just as effective when applied to societal challenges like refugee resettlement [6]. The public sector, government, human rights organizations and many entities working in humanitarian operation and non-profit logistics are also adapting to explore the challenges and opportunities of achieving social good in the age of analytics. There are many opportunities to use analytics for improving refugee resettlement decision-making processes. In the United States, after refugees are accepted to be resettled, they are assigned to a national resettlement agency which is responsible to make a decision on where to place refugees within their network of communities. Historically, this decision-making

was largely a manual process with multiple sources of inefficiencies. First, the importance of getting the *right* initial refugee-location match is evident in the literature. Many studies show that initial placement of refugees greatly affects present and future employment, education, and welfare outcomes [7–12]. However, manually estimating the welfare outcomes for refugees across communities is a challenging and complicated endeavor. Second, there is also complexity in keeping in mind all factors involved in refugee resettlement, process cases one-by-one, and find the best locations for refugees. While refugee needs should be supported by the community in which they are placed, the provision capacity of communities on various provided services should be respected as well. Moreover we are interested to best use the available resources to maximize the total expected outcomes of refugees. Analytics attempts to remedy the inefficiencies in manual refugee resettlement processes and enrich resettlement decision-making by providing optimized, data-informed recommendations on refugee-community matches. There has been a number of studies conducted on using analytics to find the optimal match between refugees and local communities. While several studies suggested preference-based matching systems for refugee resettlement [13–19], others consider measures related to refugee outcomes like estimated employment likelihood as the quality of match between refugees and localities [20]. The focus of this dissertation proposal report falls in the latter category.

The organization of this report is as follows: In Chapter 1, We first set up the static, deterministic integer optimization model for refugee resettlement that guides the matching recommendations and optimizes the total expected employment over matched refugees. In this chapter, we also explain how we estimate refugees employment probabilities from data and we discuss the backtesting we conducted to validate our approach. We implemented our algorithmic solution into an interactive decision support software, *Annie*™ Moore. Interactive visualization remains a novelty with respect to optimization outcomes that are uncertain due to accommodations needed for formulating some semblance of a model. Thus, we describe the interactive implementation and features of our software at the end of this chapter.
In Chapter 2, we study the *dynamic* nature of refugee resettlement that is, while community capacities are set on an annual basis, refugees arrive weekly and should be placed by resettlement agencies over the course of the year. We improve upon earlier formulations by incorporating the dynamic aspect of resetting refugees over time, and propose a dynamic allocation system based on two-stage stochastic programming. We provide computational results that show the employment outcomes improvement and also describe the implementation of dynamic algorithm into the interactive visualization.
In Chapter 3, we introduce the concept of *risk* in refugee resettlement optimization. We explain how the error in estimating refugee employment probabilities results in the uncertainty of their expected optimal outcome. To hedge against this risk, we propose possible approaches that explicitly incorporate risk into the optimization model and provide optimal solutions for risk-averse decision makers.

Chapter 1

Placement Optimization in Refugee Resettlement[1]

Every year tens of thousands of refugees are resettled to dozens of host countries. While there is growing evidence that the initial placement of refugee families profoundly affects their lifetime outcomes, there have been few attempts to optimize resettlement decisions. We integrate machine learning and integer optimization into an innovative software tool, *Annie* Moore, that assists a US resettlement agency with matching refugees to their initial placements. Our software suggests optimal placements while giving substantial autonomy to the resettlement staff to fine-tune recommended matches, thereby streamlining their resettlement operations. Initial backtesting indicates that *Annie* can improve short-run employment outcomes by 22%–38%. We conclude by discussing several directions for future work.

1.1 Introduction

In 2018 there were 20.4 million refugees—the highest number ever recorded—under the mandate of the United Nations High Commission for Refugees (UNHCR) [21]. Of those, the UNHCR considers 1.44 million refugees to be in need of *resettlement*—permanent relocation from their asylum country to a third country [22]. The number of cases submitted by the UNHCR for resettlement in 2018, however, was just over 81,000, with fewer than 56,000 refugees departing for resettlement [22]. Refugees in need of resettlement are particularly vulnerable: a quarter are survivors of torture and a third face persecution in their country of origin [22, Annex 3]. Currently, most refugees departing for resettlement are Syrians who seek asylum in Turkey, Lebanon, and Jordan, but there are also thousands of resettled refugees from the Democratic Republic of the Congo, Iraq, Somalia, and Myanmar.

Dozens of countries, including the United States (US), Canada, the United Kingdom (UK), Australia, France, Norway, and Sweden, resettle refugees (for refugee allocation mechanisms *across* countries, see [13] and [23]). There is ample empirical evidence that the initial place-

[1]Narges Ahani, Tommy Andersson, Alessandro Martinello, Alexander Teytelboym, and Andrew C Trapp. Placement optimization in refugee resettlement. *Operations Research*, 2021. https://doi.org/10.1287/opre.2020.2093

ment of refugees within the host countries determines their lifetime employment, education, and welfare outcomes [7–12]. Therefore, ensuring the best initial match between the refugee family and the community is crucial for social, economic, and humanitarian perspectives. Even so, resettlement capacity offered by communities is rarely being used to maximize either the welfare of refugees or of the host population.

This paper integrates machine learning and integer optimization into a software package that we call *Annie* MOORE (Matching and Outcome Optimization for Refugee Empowerment), named after Annie Moore, the first immigrant on record at Ellis Island, New York in 1892. *Annie* is, to the best of our knowledge, the first software designed for resettlement agency pre-arrival staff to recommend data-driven, optimized matches between refugees and local affiliates while respecting refugee needs and affiliate capacities. *Annie* was developed in close collaboration with representatives from all levels of the US resettlement agency HIAS (founded as the Hebrew Immigrant Aid Society), where the first version was deployed in May 2018. It was presented in August 2018 to the US Department of State and all staff at HIAS, with new features being regularly added.

We combined techniques from operations research, machine learning, econometrics, and interactive visualization to create *Annie*. The software is distinctive in that it blends rigorous analysis with careful attention to the detail of the day-to-day resettlement process for resettlement staff. As such, *Annie* integrates the generation of data-informed recommendations with substantial end-user autonomy by the end-user. This flexibility empowers staff to focus more of their resources on refugee families that might be more challenging to match (for example due to complex medical conditions). Backtesting indicates that *Annie* would have been able to increase employment outcomes among refugees resettled by HIAS in 2017 by between 22% and 38%, depending on the constraints activated by the agency staff. *Annie* also alleviates inefficiencies in the manual matching process, and holds much promise for future impact in refugee resettlement—both domestically and abroad—as well as for new applications, such as asylum matching.

The paper proceeds as follows. Section 1.2 describes the specific context of refugee resettlement in the US, and places our work in the greater context of humanitarian operations problems. Section 1.3 sets up the integer optimization model that guides the matching recommendations. In Section 1.4 we explain how we estimate counterfactual employment probabilities from data. Section 1.5 discusses the backtesting we conducted to validate our approach. Section 1.6 describes the implementation and features of our software, while Section 1.7 concludes and points to many directions for further work. Appendices include detailed data descriptions, estimation procedures and diagnostics, and a variety of experiments using different objective functions and testing the sensitivity of our modeling.

1.2 Background Context and Previous Work

Because *Annie* helps HIAS to resettle refugees in the US, we briefly describe the US resettlement program.

1.2.1 Refugee Resettlement in the United States

The US has historically been, by a wide margin, the world's largest destination of resettled refugees, with 22,491 arriving in 2018, down from 53,716 in 2017, and 84,994 in 2016 (in this

manuscript, all references to years of refugee data are presented in terms of the *fiscal* year, that is from October 1 through September 30) [24]. In terms of refugees resettled per capita (calendar year), the US trails a number of countries including Sweden, Norway, Canada, New Zealand, Iceland, and Australia. The US Refugee Admissions Program (USRAP) resettles US refugees and is managed by the Bureau of Population, Refugees and Migration (PRM) of the US Department of State, with the assistance of the US Citizenship and Immigration Services (USCIS) of the US Department of Homeland Security, and the Office of Refugee Resettlement (ORR) of the US Department of Health and Human Services (HHS). Alongside the UNHCR and the International Organization for Migration, these agencies coordinate identifying refugees, conducting security checks, and arranging for travel funding from the refugees' destinations.

The actual matching of refugees to their initial placements is delegated to nine resettlement agencies, previously known as *voluntary agencies*. In addition to HIAS, these agencies include Church World Service (CWS), Ethiopian Community Development Council (ECDC), Episcopal Migration Ministries (EMM), International Rescue Committee (IRC), Lutheran Immigration and Refugee Service (LIRS), US Conference of Catholic Bishops (USCCB), US Committee for Refugees and Immigrants (USCRI), and World Relief (WR). HIAS handles around 5% of all refugees in the US, resettling 2,038 refugees in 2017 and 3,844 refugees in 2016. The resettlement agencies are responsible for developing their own networks of *affiliates* in local communities that welcome refugees and help them integrate into a new life in the US. Affiliates offer resettlement capacity voluntarily, although affiliate capacity is monitored and approved by the US government. There are currently around 360 affiliates in approximately 200 local communities across the US, and HIAS operates 20 of them at the time of this writing.

Resettlement agencies match refugees to affiliates during the resettlement process largely by hand. Resettlement staff from each agency meet weekly to select, in round-robin fashion, from a pool of "cleared for arrival" refugee *cases*. Each case consists of an immediate *family* of one or more members (we use *case* and *family* interchangeably). A significant portion (roughly one third) of these are free cases, that is, they have no relatives in the US. Such cases are especially vulnerable, as the absence of family support exacerbates the challenges of lacking language skills and independent financial means. Thus, the responsible agency must carefully leverage its affiliate network to inform their case selection. After each agency selects their set of weekly cases, staff manually assess—on a one-by-one basis—the feasibility and fit of cases to locations in their network. In addition to integration factors such as language and nationality feasibility, the fit between the affiliate and the family depends on various community capacities, such as available placement capacity, housing availability, slots for English language instruction, and employment prospects.

This manual process creates multiple inefficiencies that motivated the development of *Annie*. First, it is organizationally demanding for HIAS staff to keep in mind various support attributes such as languages, nationalities, family composition, and medical needs for all affiliates. This information overload at times results in not meeting the needs of refugees and in stretching the provision capacity of the affiliates. Second, while established indicators exist to assess the degree to which a refugee has successfully integrated into their new surroundings, estimating and optimizing these welfare outcomes manually is prohibitive. Established indicators include employment and economic sufficiency, developed social networks, and civic engagement activities like voting [see, e.g., 25, 26]. Hence, refugees are often not placed to the best available affiliate even according to well-defined outcome metrics. Third, inefficiencies arise from processing refugees case-by-case, in sequential fashion, rather than matching all

arriving refugees to affiliates simultaneously. We show that *Annie* resolves or mitigates each of these inefficiencies.

1.2.2 Related Literature

Our work builds on a number of contemporary studies in humanitarian matching systems. One recent example is a tool to match children in state custody to families for adoption used by the Pennsylvania Adoption Exchange [27]. Bansak et al. [20] first proposed to use machine learning and linear programming for refugee resettlement based on employment data from the US and Switzerland. Using a similar dataset to theirs, we expand on their estimation techniques, while extending their optimization methods. Our integer optimization model extends the multiple multidimensional knapsack model for refugee matching (see also Delacrétaz et al. [18], Trapp et al. [28], and Nguyen et al. [19]). However, as we focus on outcome optimization, our work differs substantially from papers that suggested preference-based matching systems for refugee resettlement [13–19].

Placement optimization in refugee resettlement shares many common features with other problems in humanitarian operations [29, 30]. Typical challenges in this sector include severe lack of resources—financial, labor, time, and data—as well as complex decision environments. The refugee resettlement decision environment includes refugees as well as local communities, non-profit organizations, donors, and federal, state and local governments. Hence, similar to other humanitarian operations problems, placement optimization also diverges from the traditional stance of optimizing a single financial metric, and may consider alternative objectives such as those based on equity [see, e.g., 31–33]. Refugee resettlement is perhaps most differentiated by its particular exposure and sensitivity to shifting political climates and attitudes, both domestic and abroad. This volatility generates significant uncertainty with respect to the operating and planning environments of resettlement agencies.

1.3 Integer Optimization for Refugee Resettlement

We formulate the operational challenge of matching refugee families to local communities, or affiliates, presently solved manually by resettlement agencies, using mathematical optimization.

1.3.1 Formal Problem Setup

We use i, j, k, and ℓ as indices for family (case), member, service and affiliate, respectively. For any placement period, let $\mathcal{F} = \{F^1, F^2, \ldots, F^i, \ldots, F^{|\mathcal{F}|}\}$ be the set of refugee families to be placed. Each family F^i is a set of refugees consisting of one or more members, $F^i = \{f^{i,1}, f^{i,2}, \ldots, f^{i,j}, \ldots f^{i,|F^i|}\}$. For clarity of exposition, we refer to member j of family F^i as f^{ij}. Denote as N_w^i the set of working-age refugees in family F^i, where $N_w^i \subseteq F^i$. Denote the set of all refugees as $\mathcal{R} = \cup_{i\in\{1,2,\ldots,|\mathcal{F}|\}} \cup_{j\in\{1,2,\ldots,|F^i|\}} \{f^{ij}\}$. Moreover, let the set of affiliates (localities) to which families are resettled be $\mathcal{L} = \{L^1, L^2, \ldots, L^\ell, \ldots L^{|\mathcal{L}|}\}$.

A family F^i requires various *capacitated services* from a set $\mathcal{S} = \{S^1, S^2, \ldots, S^k, \ldots S^{|\mathcal{S}|}\}$. The needs of family F^i are summarized by vector s^i, where s_k^i denotes the required units of service k. The set $\mathcal{S}$ may include services such as raw weekly refugee processing capacity at affiliates, slots in foreign language instruction (such as ESL), school seats for children in the

family, and housing availability. For every service S^k of local affiliate L^ℓ, at most $\bar{s}_k^\ell$ units may be filled by families placed in affiliate L^ℓ. There may also be a requirement of at least $\underline{s}_k^\ell$ units of the service S^k to be filled by the families placed in affiliate L^ℓ (we assume $\underline{s}_k^\ell \leq \bar{s}_k^\ell$); in practice, nonzero lower bounds exist for certain services, such as ensuring regular, positive refugee placement in affiliates.

For every family F^i and local affiliate L^ℓ, let binary variable z_ℓ^i equal 1 if family F^i is matched to affiliate L^ℓ, and 0 otherwise. Let $a_\ell^i \in \{0,1\}$ indicate whether family F^i can be feasibly placed in affiliate L^ℓ. The value of a_ℓ^i is a priori determined by evaluating the compatibility of family F^i with various binary community support services at affiliate L^ℓ, such as language and nationality, as well as large family and single parent support conditions (should these be present in the family). We denote these community support services as *binary services*.

We attribute to each refugee-affiliate match a single number called the *quality score*. The function $q : \mathcal{R} \times \mathcal{L} \to \mathbb{R}_{\geq 0}$ defines quality score q_ℓ^{ij} for any $f^{ij} \in \mathcal{R}$ and any $L^\ell \in \mathcal{L}$. We are interested in the scenario where q represents the employment outcome of refugee f^{ij} in affiliate L^ℓ and can be estimated from data using observable affiliate and family characteristics. In Section 7 we discuss the sole use of employment data to generate these estimates (indeed, no other data related to integration outcomes is systematically available). We aggregate the refugee level quality scores q_ℓ^{ij} of each family F^i and affiliate ℓ into a case-level *value* (or weight) v_ℓ^i. The primary means of aggregation that we consider is the *sum* of individual scores over each family $v_\ell^i = \sum_{j=1}^{|F^i|} q_\ell^{ij}$ (SUM). Discussions on alternative interpretations of case-level quality scores can be found in Appendix D.

1.3.2 Placement Optimization

We now present integer optimization problem RefMatch, represented by (1.1a)–(1.1e):

$$\text{maximize} \quad \sum_{i=1}^{|\mathcal{F}|} \sum_{\ell=1}^{|\mathcal{L}|} v_\ell^i z_\ell^i \tag{1.1a}$$

$$\text{subject to} \quad \sum_{\ell=1}^{|\mathcal{L}|} z_\ell^i \leq 1, \ \forall\, i, \tag{1.1b}$$

$$\underline{s}_k^\ell \leq \sum_{i=1}^{|\mathcal{F}|} s_k^i z_\ell^i \leq \bar{s}_k^\ell, \ \forall\, \ell, \ \forall\, k, \tag{1.1c}$$

$$z_\ell^i \leq a_\ell^i, \ \forall\, i, \ \forall\, \ell, \tag{1.1d}$$

$$z_\ell^i \in \{0,1\}, \ \forall\, i, \ \forall\, \ell. \tag{1.1e}$$

Objective function (1.1a) maximizes the total value over all matched families to affiliates. Constraint set (1.1b) ensures that families are placed in at most one affiliate. Constraint set (1.1c) ensures that lower and upper bounds are respected for all capacitated services and affiliates. Constraint set (1.1d) ensures that family-affiliate matches can only occur when the affiliate can support the needs of the family, that is, the necessary binary services exist. Variable domains are specified in (1.1e). Finally, let $z^\star$ be the optimized match outcome, that is, the optimal solution representing the assignment of families to affiliates that optimizes objective function (1.1a).

While integer optimization problem RefMatch bears similarity to a variety of knapsack-like problem classes, we are unaware of another application of this particular form:

- When $|\mathcal{S}| = 1$, $\underline{s}_k^\ell = 0 \; \forall \; \ell$, and $s_k^i = 1 \; \forall \; i$, the optimization problem can be solved via linear programming [20].
- When $|\mathcal{S}| = 1$ and $\underline{s}_k^\ell = 0 \; \forall \; \ell$, it is the NP-hard *multiple 0–1 knapsack problem* which features multiple knapsacks and items that consume integer resources for the knapsack in which they are placed [34].
- When $|\mathcal{L}| = 1$ and $\underline{s}_k^\ell = 0 \; \forall \; k$, it is the NP-hard *multidimensional 0–1 knapsack problem* which features knapsack items that consume integer resources along multiple dimensions [35].
- When $\underline{s}_k^\ell = 0 \; \forall \; \ell, k$, it is the NP-hard *multiple multidimensional knapsack problem* and combines features of both, that is, multiple knapsacks along multiple dimensions [36]. If in addition, $\sum_{\ell=1}^{|\mathcal{L}|} z_\ell^i = 1 \; \forall \; i$, it is the NP-hard *multiple-choice multidimensional knapsack problem* [37]; in our setting, there is no requirement (in theory) for every family to be placed in an affiliate.

Integer optimization problem RefMatch generalizes the *multiple multidimensional knapsack problem* of [36], as it allows for positive lower bounds $\underline{s}_k^\ell$ for any services and affiliates. The existence of such lower bounds differentiates it from the multiple multidimensional knapsack problem, as it may lead to infeasibility. The formulation is valid over any operational period. Due to its generality, our model can be customized to specific refugee resettlement settings. Section 1.5 shows the results of testing the sensitivity of our model under three different scenarios. First, we test the effect of relaxing upper bounds (1.1c) for the number of total resettled refugees. Second, we test the effects of lower bounds (1.1c) expressed as distributional requirements (such as minimum average case sizes across affiliates) and as lower bounds on the total number of resettled refugees. We also consider the effects of relaxing the binary service constraints (1.1d). We discuss alternative models and objective functions and conduct sensitivity checks in Appendices D, E, and F.

1.4 Estimation of Counterfactual Employment Probabilities

We use the estimated probability of employment of refugee f^{ij} in each affiliate L^ℓ as a measure of quality score, or:

$$q_\ell^{ij} = E\left[y_{ij} \mid \mathbf{X}_{ij}, \ell\right],$$

where y_{ij} is (binary) outcome data indicating employment status of refugee f^{ij} within 90 days of arrival in the United States, and $\mathbf{X}_{ij}$ is a set of observable refugee characteristics and quarterly macroeconomic variables. We use national employment ratio and unemployment rate as macroeconomic variables, which are common to all refugees arriving in a given quarter. Further details on the available data appear in Appendix A.

Using expected potential outcomes rather than stated preferences for our counterfactual analysis creates two challenges. First, y_{ij} is unobserved for incoming refugees. Second, even

for past refugees we only observe $y_{ij}|x^{ij}_{\ell^\star}$, that is, employment status of refugee f^{ij} in $\ell^\star$, the affiliate to which they were actually assigned in the data. We do not observe the corresponding potential outcome distribution $y_{ij} \mid x^{ij}_{\ell} \; \forall \; \ell \neq \ell^\star$. Moreover, the functional form connecting y_{ij}, $\mathbf{X}_{ij}$, and ℓ is unknown. Specific synergies may exist between refugee characteristics and affiliates that affect refugee integration. Following [20], we thus exploit machine learning approaches to compute $\hat{q}^{ij}_{\ell}$, the estimated probability of employment of refugee f^{ij} in affiliate L^{ℓ}. Using data on refugees arriving between 2010 and 2016, we estimate both semi- and non-parametric functions $\hat{f}_{\ell} : \mathcal{R} \rightarrow \mathbb{R}_{\geq 0}$ such that $\hat{q}^{ij}_{\ell} = \hat{f}_{\ell}(\mathbf{X}_{ij})$. We then test the performance of these models on refugees arriving in 2017.

In the estimation process we only use free cases, which are those refugees (individuals or families) that the resettlement agency can assign to any of the affiliates. We therefore exclude refugees with pre-existing family ties, which are almost always pre-assigned to the affiliate where their pre-existing connection resides. This choice restricts the samples we use to train and test the models to 2,486 and 498 refugees, respectively.

While it may be tempting to increase the number of available observations for model estimation by including all refugees resettled by HIAS, the additional refugees will likely differ from the free cases to which *Annie* will be applied, and including them in the estimation might introduce bias and likely overestimate existing synergies for free cases. For example, because of pre-existing networks, family reunifications enjoy particular advantages [38, 39] that would bias our estimates. By restricting our sample to free cases, we align the sample used for estimation with the sample on which *Annie* will be applied.

We estimate effects on employment for the seven (out of twenty) affiliates receiving at least 200 refugees up to 2016, and aggregate the remaining affiliates into a single partition ℓ_0. In a parametric approach, it is possible to estimate a fully saturated logit model for employment where flexible transformations of refugee characteristics $\mathbf{X}_{ij}$ are interacted with $\ell - 1$ affiliate dummies. Such an approach would, however, estimate an overly complex model, with poorly identified coefficients, and therefore yield poor predictive properties.

We thus use two alternative machine learning models. First, we introduce a Least Absolute Shrinkage and Selection Operator (LASSO) constraint to the interacted logit model to reduce model complexity. The single LASSO hyper-parameter disciplines both main and interaction terms with the same weight, biasing them towards zero (and thus biasing predictions towards the mean). Second, we follow [20] and estimate a Gradient Boosted Regression Tree (GBRT), an iterative ensemble of classification trees. We set the hyper-parameters of these models via 5-fold cross-validation on our training sample (we internally calibrate constraint strength for LASSO, and the learning rate and pre-pruning level for GBRT). We choose hyper-parameter values for each model by maximizing the area under a Receiver Operating Characteristic (ROC) curve.

We benchmark both models against the performance of a naïve constant estimator [see, e.g., 20], as well as two second-best standards. The first benchmark model is a standard logit model that includes all variables in $\mathbf{X}_{ij}$, but does not attempt to estimate affiliate-specific effects. The second benchmark model is a logit model with no LASSO constraint, where $\mathbf{X}_{ij}$ interacts with all ℓ affiliates. Table 1.1 shows that both LASSO and GBRT outperform the second-best benchmarks by over 20% in terms of misclassification error when applied to 2017 refugees. With respect to the constant-logit benchmark used by [20] we obtain a 37% and 34% improvement using LASSO and GBRT respectively, which is comparable to the 28% they obtain in their US data. The area under the ROC is highest for LASSO, but overall both models exhibit

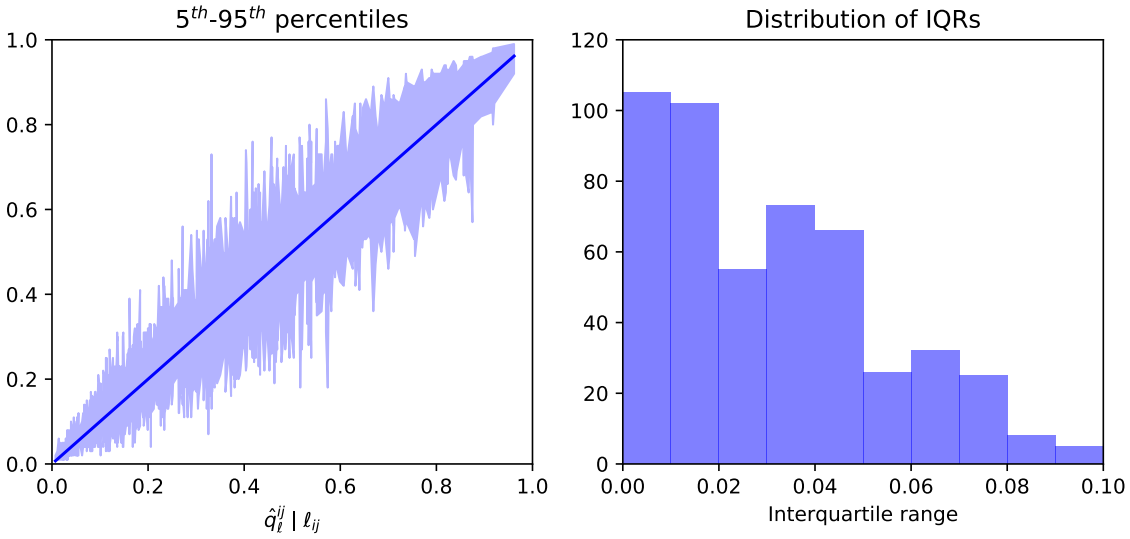
	Training data	Test data			
	Misc. error	Misc. error	Recall (1)	Precision (1)	AUC-ROC
Constant	0.259	0.319	0.000	0.000	0.500
Logit	0.240	0.259	0.503	0.615	0.790
Logit (by affiliate)	0.172	0.283	0.541	0.558	0.769
LASSO	0.161	0.203	0.434	0.622	0.797
Gradient boosted tree	0.099	0.205	0.396	0.606	0.795

NOTE: *Misclassification error* is the proportion of observations incorrectly classified. *Recall* measures the proportion of correctly predicted employed refugees among refugees actually employed (true positives over true positives plus false negatives). *Precision* measures the proportion of correctly predicted employment cases among all predicted employment cases (true positives over true positives plus false positives). All of these measures refer to a binary classification with a threshold set at the standard value of 0.5. Because our measure of quality scores uses predicted probabilities of employment, this specific threshold does not affect optimal allocations. *AUC-ROC* measures the area under the Receiver Operating Characteristic Curve for each model (ROC curves appear in Appendix B).

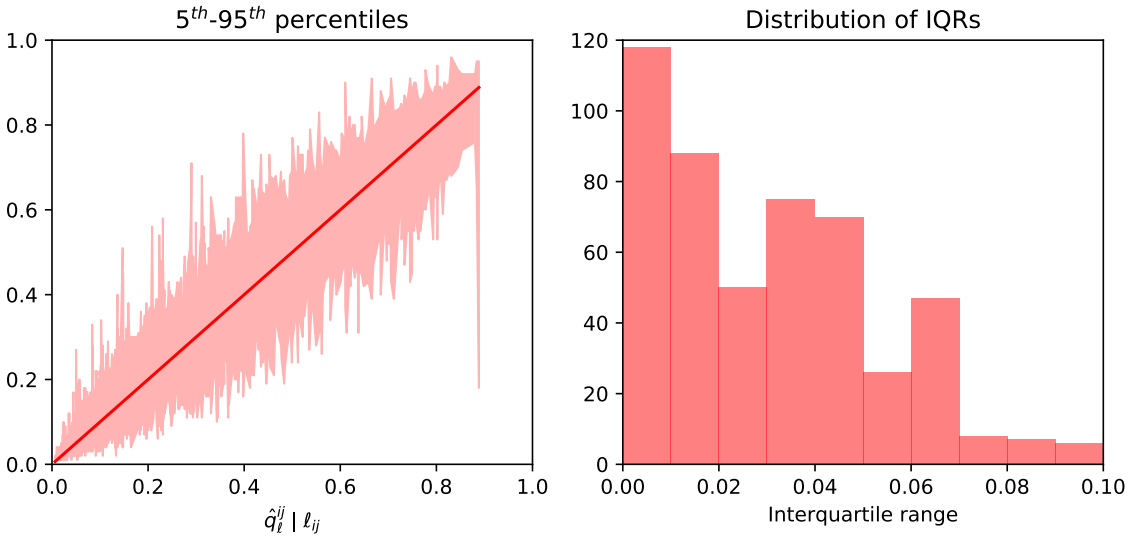
Table 1.1: Model performance.

similar predictive power. LASSO, however, produces slightly more stable and well-calibrated predictions, particularly for observations with high predicted employment probabilities. We obtain these results by bootstrapping the distribution of predictions for each data point in the test set given assignment of refugee f^{ij} to $\ell^\star$. In each of a thousand iterations, we re-sample with replacement the training dataset, re-estimate each model and compute a new predicted probability of employment. The right panels of Figure 1.1 show the 5^{th} to 95^{th} percentiles of the prediction distributions for each data point in the test sample. The left panels show the distribution of bootstrapped interquartile ranges for each data point.

LASSO tends to produce more narrow predictions for refugees with high baseline probability of employment, which are highly relevant for the quantification of employment gains. LASSO is also better calibrated than GBRT—with 159 employed refugees in our test set, whereas the sum of predicted employment probabilities given assignment of refugee f^{ij} to $\ell^\star$ is 157.93 for LASSO, it is only 142.96 for GBRT (calibration plots appear in Appendix B). Thus, while using either model has very similar consequences for optimal refugee assignment, in the remainder of the paper we quantify employment gains given the quality scores predicted by LASSO (and in Appendix C, we replicate employment gains under the predictions of GBRT).



(a) LASSO



(b) Gradient Boosted Regression Tree (GBRT)

Figure 1.1: Bootstrapped uncertainty of predicted employment probabilities in 2017 for LASSO and GBRT model. Left panels: prediction distributions (5th-95th percentile) for each data point in test sample. Right panels: distribution of interquartile ranges for each data point in test sample.

1.5 Counterfactual Optimization Outcomes

We now describe the counterfactual impact of using our integer optimization problem REFMATCH. We create test scenarios that result from varying three constraint sets. To quantify the impact of optimally reassigning refugees to affiliates, we use the employment probabilities for each affiliate estimated in Section 1.4. We compute the counterfactual gain in employment relative to our prediction from the LASSO model for 2017. Since our prediction is very close to the actual employment values—the LASSO model predicts 157.93 employed refugees versus 159 who were actually employed in the testing data—our optimization is a meaningful counterfactual exercise.

Our objective function (1.1a) maximizes the total expected number of employed refugees. Our binary service constraints (1.1d) are: language, nationality, single-parent, and large-family support. We set the capacity constraints (1.1c) for each affiliate relative to the observed capacity in 2017. Moreover, we specify minimum average case sizes to enforce distributional constraints via the lower bounds in (1.1c). We vary the following three factors to create our test scenarios.

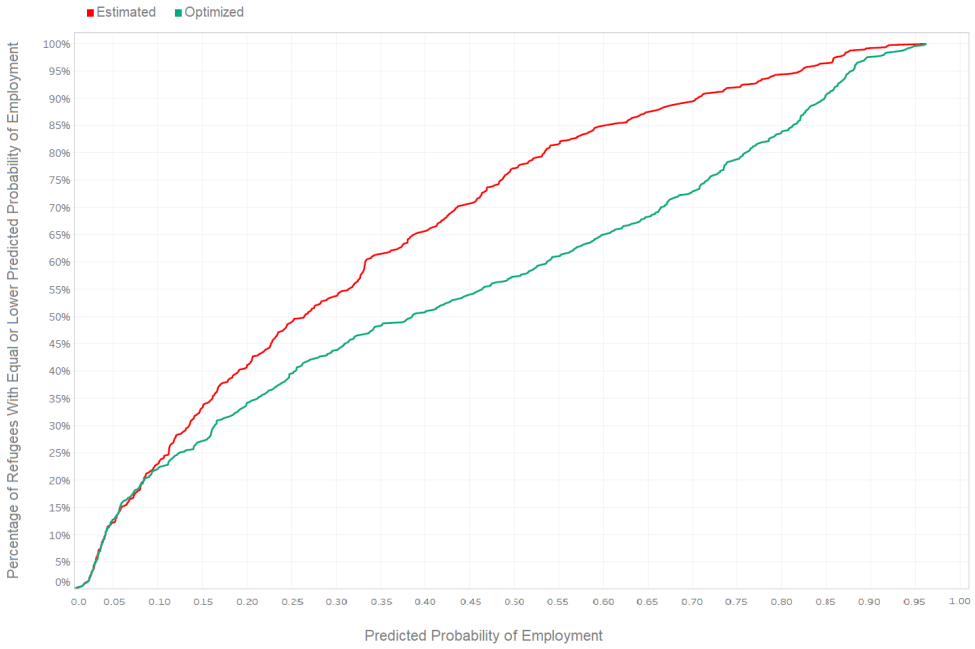
Affiliate capacity. Affiliate capacity is federally approved, but can be exceeded by up to 10% without further pre-approval. Moreover, a common aim of the agencies is to fill at least 90% of the approved capacity at each affiliate. In 2017, somewhat unusually, approved capacity was much higher than the observed number of arriving refugees. We therefore use the observed placements at each affiliate to set sensible counterfactual capacities. We test three values: {observed capacity with no lower bound; 110% of the observed capacity with no lower bound; and 110% of observed capacity with a lower bound of 90% of observed capacity}.

Binary service constraints. In the observed 2017 placements, binary service constraints were violated 38 times (26 language constraints, 1 nationality constraint, 8 single-parent constraints, and 3 large-family constraints), representing approximately 12% of resettled cases. However, binary service constraints, especially language constraints, can be important to ensure successful refugee integration. We therefore test two values: {binary service constraints are activated (ON), binary service constraints are not activated (OFF) }.

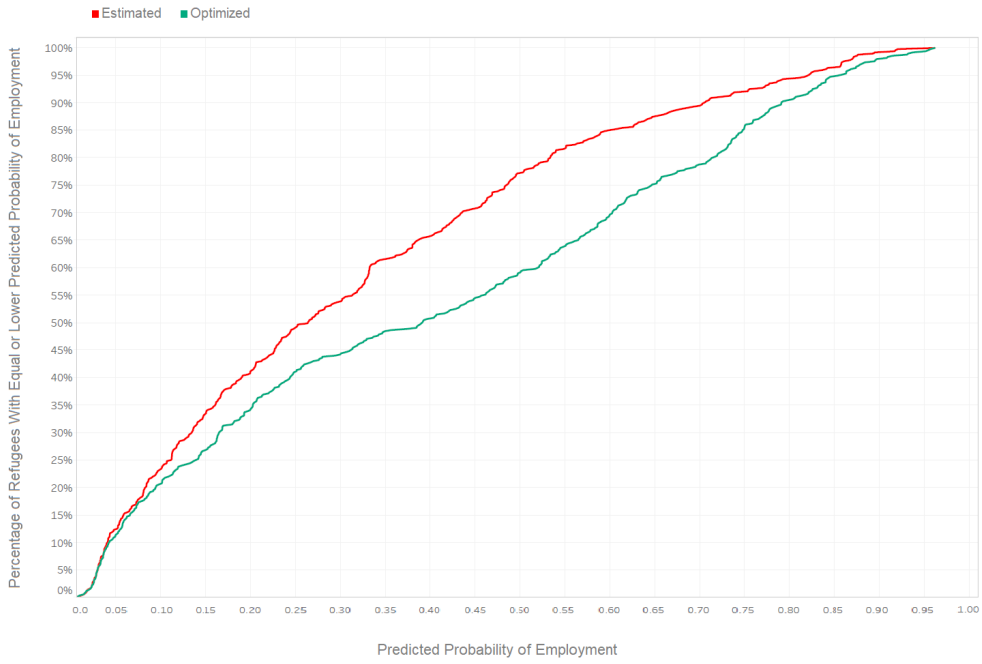
Minimum average case size in each affiliate. A placement that maximizes the total expected number of employed refugees could potentially pack many single-refugee cases or large-family cases into the same affiliate. This could be seen as unfair by the agencies, reduce support for resettlement, and stymie refugee integration. Therefore, to capture such equity considerations, we experiment with the implementation of a minimum average case size in each affiliate. The average case size in our 2017 test dataset across all affiliates is 2.55. We therefore test five values: {no minimum average case size, observed minimum average case size at each affiliate, 2, 2.5, 3}.

In total, we have $3 \times 2 \times 5 = 30$ counterfactual test scenarios. Akin to [20], we conduct our experiments using capacity levels for the period of one year. All experiments were run on a laptop computer with an Intel(R) Core(TM)i5-8365U 1.60GHz processor and 16GB RAM running 64-bit Microsoft Windows 10 Enterprise. The Gurobi Optimizer v9.0.0 [40] and Python 3.7.4 was used for all counterfactual optimization testing in Section 1.5, and the optimality gap tolerance parameter MIPGap was set to 0. We summarize our results in Table 1.2.

First, note that without minimum average case size constraints, the gain in employment from optimization is over 30% in all scenarios. As Figures 1.2a and 1.2b show, the employment probability distribution after optimization (almost) first-order stochastically dominates the



(a) Cumulative distribution of employment probabilities. Red: estimated probabilities under HIAS placement. Green: optimized probabilities for {observed capacity, activated binary service constraints, no minimum average case size} scenario.



(b) Cumulative distribution of employment probabilities. Red: estimated probabilities under HIAS placement. Green: optimized probabilities for {observed capacity, activated binary service constraints, at least observed average case size} scenario.

Figure 1.2: Employment gains from optimizing refugee placement.

Capacity Adjustment	Min Avg Case Size	Binary Service Constraints	Total Expected Employed Refugees	Gains wrt to Predicted Employed Refugees (157.93)	St Dev in Avg Case Size Across Affiliates	# of Unplaced Cases / Refugees	#of Affiliates Violating 90% Capacity	# and % of Cases/Refugees Violating Constraints	Build and Run Time (s)
Observed	None	Off	213.02	34.89%	1.29	0/0	0	70/209 (21.28%/24.91%)	0.48
Observed	None	On	208.25	31.86%	1.35	3/10	1	0/0 (0.00%/0.00%)	0.36
Observed	2	Off	206.28	30.61%	1.04	1/1	0	81/220 (24.62%/26.22%)	0.92
Observed	2	On	202.03	27.92%	1.17	2/9	1	0/0 (0.00%/0.00%)	0.93
Observed	2.5	Off	196.76	24.59%	0.33	1/1	0	97/265 (29.48%/31.59%)	5.43
Observed	2.5	On	192.95	22.17%	0.14	3/7	0	0/0 (0.00%/0.00%)	3.87
Observed	3	Off	172.83	9.43%	0.65	78/86	6	71/217 (21.58%/25.86%)	7.78
Observed	3	On	169.64	7.42%	0.65	79/89	6	0/0 (0.00%/0.00%)	4.97
Observed	Observed	Off	199.34	26.22%	0.84	2/2	0	81/232 (24.62%/27.65%)	5.40
Observed	Observed	On	195.65	23.89%	1.09	4/8	1	0/0 (0.00%/0.00%)	2.80
$\leq 110\%$	None	Off	218.06	38.07%	1.40	0/0	1	71/199 (21.58%/23.72%)	0.78
$\leq 110\%$	None	On	212.96	34.84%	1.42	2/9	2	0/0 (0.00%/0.00%)	0.55
$\leq 110\%$	2	Off	212.39	34.48%	1.16	0/0	2	75/226 (22.80%/26.94%)	1.09
$\leq 110\%$	2	On	207.72	31.53%	0.95	2/9	3	0/0 (0.00%/0.00%)	0.85
$\leq 110\%$	2.5	Off	202.75	28.38%	0.38	0/0	1	87/222 (26.44%/26.46%)	5.19
$\leq 110\%$	2.5	On	198.84	25.90%	0.66	3/7	3	0/0 (0.00%/0.00%)	3.23
$\leq 110\%$	3	Off	177.51	12.40%	0.00	78/86	5	65/191 (19.76%/22.77%)	5.66
$\leq 110\%$	3	On	174.27	10.34%	0.90	79/89	6	0/0 (0.00%/0.00%)	3.58
$\leq 110\%$	Observed	Off	204.27	29.34%	0.83	0/0	3	81/207 (24.62%/24.67%)	6.35
$\leq 110\%$	Observed	On	200.49	26.95%	1.07	3/7	4	0/0 (0.00%/0.00%)	8.03
$[90\%, 110\%]$	None	Off	218.06	38.07%	1.36	0/0	0	68/189 (20.67%/22.53%)	1.23
$[90\%, 110\%]$	None	On	212.91	34.82%	1.14	1/2	0	0/0 (0.00%/0.00%)	1.01
$[90\%, 110\%]$	2	Off	212.39	34.48%	0.95	0/0	0	72/194 (21.88%/23.12%)	1.40
$[90\%, 110\%]$	2	On	207.58	31.44%	1.05	2/6	0	0/0 (0.00%/0.00%)	1.67
$[90\%, 110\%]$	2.5	Off	202.75	28.38%	0.32	0/0	0	79/198 (24.01%/23.60%)	6.57
$[90\%, 110\%]$	2.5	On	198.81	25.89%	0.61	2/3	0	0/0 (0.00%/0.00%)	5.95
$[90\%, 110\%]$	3	Off			Infeasible instance				
$[90\%, 110\%]$	3	On			Infeasible instance				
$[90\%, 110\%]$	Observed	Off	204.26	29.34%	0.86	5/5	0	74/200 (22.49%/23.84%)	24.71
$[90\%, 110\%]$	Observed	On	200.36	26.86%	1.10	6/7	0	0/0 (0.00%/0.00%)	12.60

Table 1.2: Results of counterfactual employment optimization under various scenarios using the SUM objective and LASSO model.

pre-optimized estimated distribution. Therefore, the estimated probabilities of employment increase for all refugees after optimization. Moreover, Figure 1.3 shows that employment rates rise in nearly two-thirds of the affiliates after optimization. Table 1.2 further indicates that, if we do not impose binary service constraints, they are violated for around a quarter of the refugees—a rate much higher than in the test data (approximately 12%). However, the presence of binary service constraints and of increasing capacity has a fairly small impact on employment gains. Indeed, because in some cases our model leaves some refugees unplaced (meaning that they would need to be placed manually by agency staff), our employment gain estimates should be even higher.

However, in these scenarios the optimization suggests rather unequal placement. Figure 1.4 compares the distribution of average case sizes in each affiliate to the distribution under our second counterfactual optimization which produces the largest variance in average case sizes. Figure 1.5a shows that without distributional constraints, many single-person cases are placed in just three affiliates that offer a high probability of obtaining employment to many types of refugees. Other affiliates get much larger cases on average. This allocation may not be acceptable to a resettlement agency. Thus, we evaluated the placement optimization by enforcing minimum average case size constraints. At low values (up to 2.5) and at observed 2017 average case size values, the optimization is still able to realize employment gains of well over 20% (see also Figure 1.5b). This is extremely encouraging because it shows that our optimization performs well even under tight distributional constraints. However, at high average case sizes, the constraints bind harder and either reduce the performance of the model substantially (by not placing many refugees), or simply cause infeasibility. It should be noted that these are precisely the instances for which many cases and refugees are unplaced, thus causing reduced optimal objective function values and corresponding gains.

We report the runtime as the time (in seconds) to both build the optimization model and solve it to global optimality using Gurobi [40]. It can be immediately observed that for the

entire FY17 dataset (839 refugees / 329 cases / 498 working-age refugees / 20 affiliates), the combined build and solve times in Gurobi finish in well under one minute (actually, under 30 seconds), with a median combined runtime of less than four seconds.

Overall, our optimization produces a substantial gain in employment, ensures that refugee binary services are better satisfied, and important distributional considerations can be respected. Moreover, the resettlement agency may impose any subset of the binary service constraints, or introduce constraints on the number of refugees with certain regional origins (although regional constraints were formerly officially considered in US placements, they are no longer specified).

It is worth emphasizing that the space of objective functions and constraints that the resettlement agency can impose within our model is much richer than what we have presented here. For example, the agency could select a different employment objective function, such as maximizing the sum of *maximum* employment probabilities within every matched case. In Appendix D we provide further experiments that optimize over several reasonable (including equity-based) measures based on derived from the individual refugee-level quality scores q_ℓ^{ij}; we find that all perform fairly well with respect to gains in employment. Further details on these experiments can be found in Appendix D.

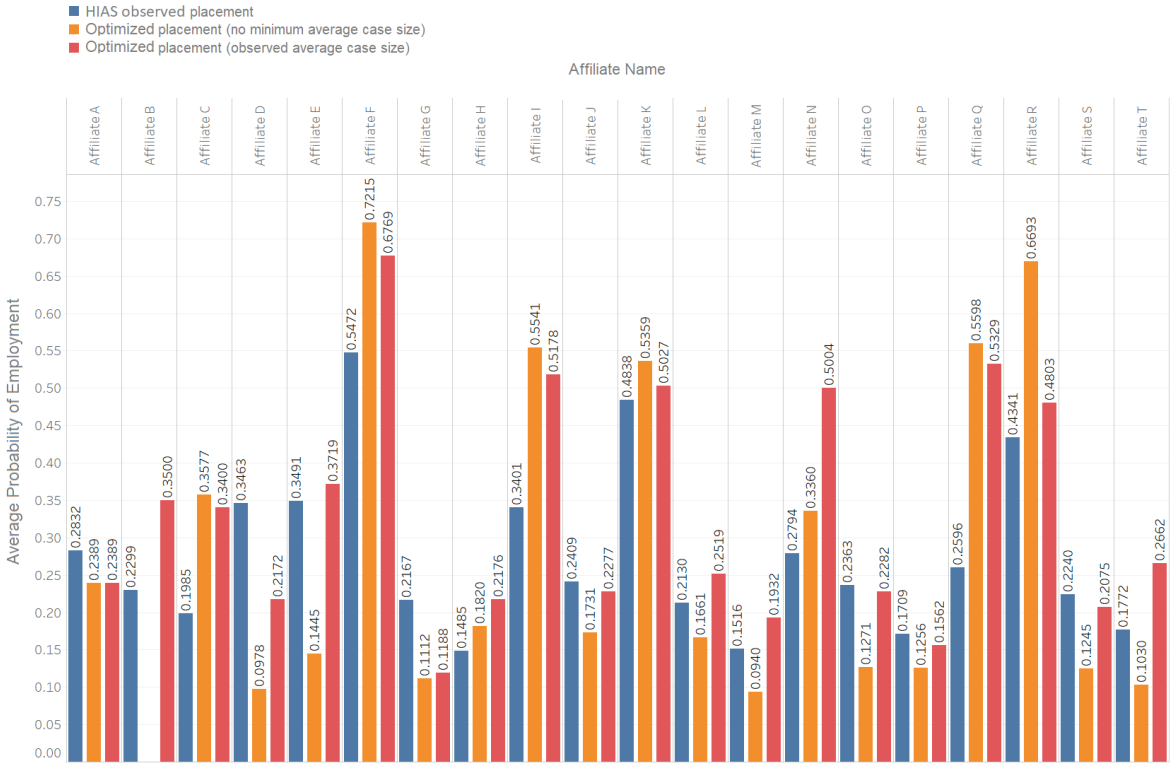


Figure 1.3: Average probability of employment at each affiliate. Blue bar: estimated probabilities under HIAS placement. Orange bar: average probability of employment for observed capacity, activated binary service constraints, no minimum average case size scenario. Red bar: average probability of employment for {observed capacity, activated binary service constraints, at least observed average case size} scenario.

We also recognize that there is inherent uncertainty in the modeling environment with respect to estimating the quality score $\hat{q}_\ell^{ij}$ for each f^{ij} in affiliate L^ℓ. In Appendix E, we inves-

tigate how the objective function changes under our optimized placement outcome $z^\star$ when we resample $\hat{q}_\ell^{ij}$ from the estimated distribution. In particular, we observe that the refugee allocation determined by our approach produces stable employment gains, and that these employment gains are not artificially inflated by uncertainty in the estimation of employment probabilities. The average expected employment given uncertainty in our predicted probabilities is within 2% of that obtained in our original backtesting for almost all of the considered scenarios.

Finally, we note that these outcomes were obtained by optimizing placement of all refugees in FY17 without splitting into multiple periods, that is, over the entire year ($n = 1$), on par with experiments reported in [20]. While desirable, experiments with $n > 1$ placement periods in a given year introduced some additional nuances that required equally detailed implementation strategies. Even so, we present such experiments in Appendix F. The key takeaways include that increasing the number of periods to $n \in \{4, 12, 52\}$ (that is, quarterly, monthly, and weekly) for placing refugees, and thereby allowing for the innate arrival stochasticity present in FY17 data, reveal encouraging results. While gains are indeed largest for $n = 1$, our methods perform very well for $n = 4$ and $n = 12$, and respectably even for $n = 52$. We refer the reader to Appendix F for additional details.

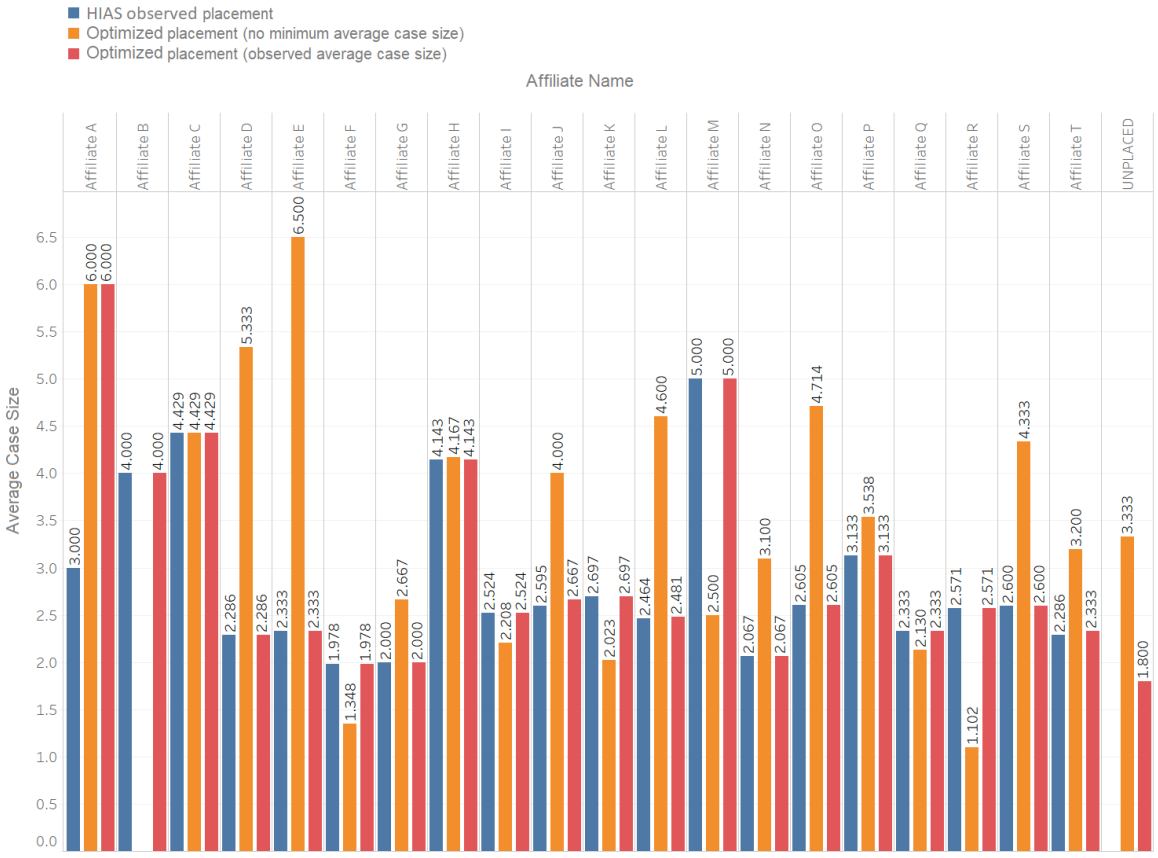
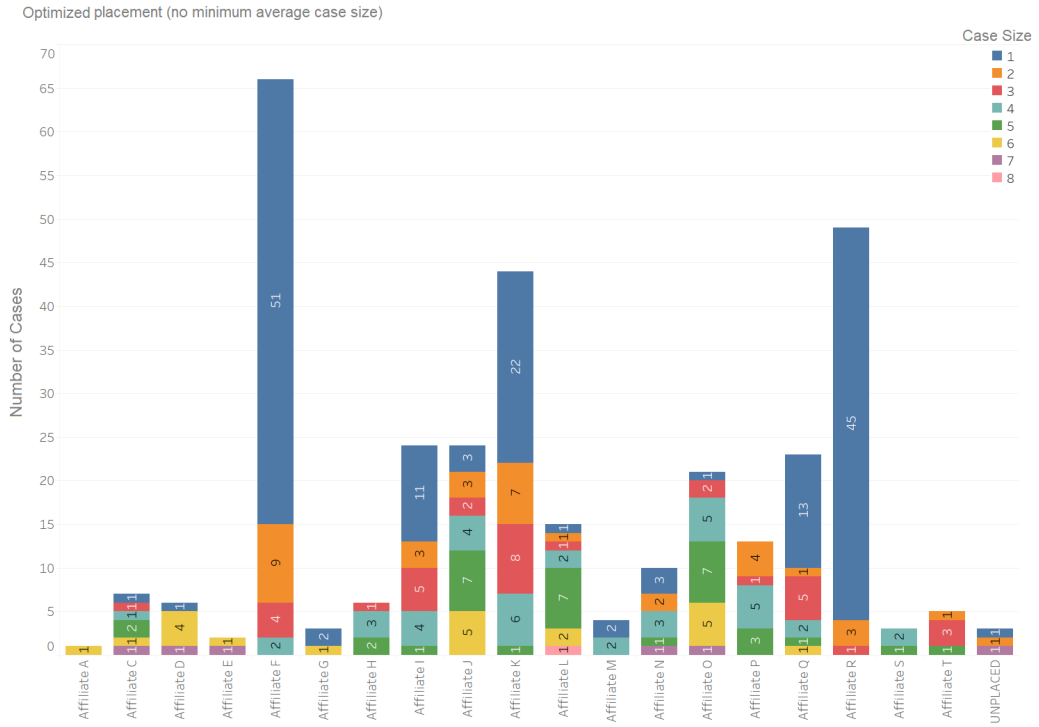
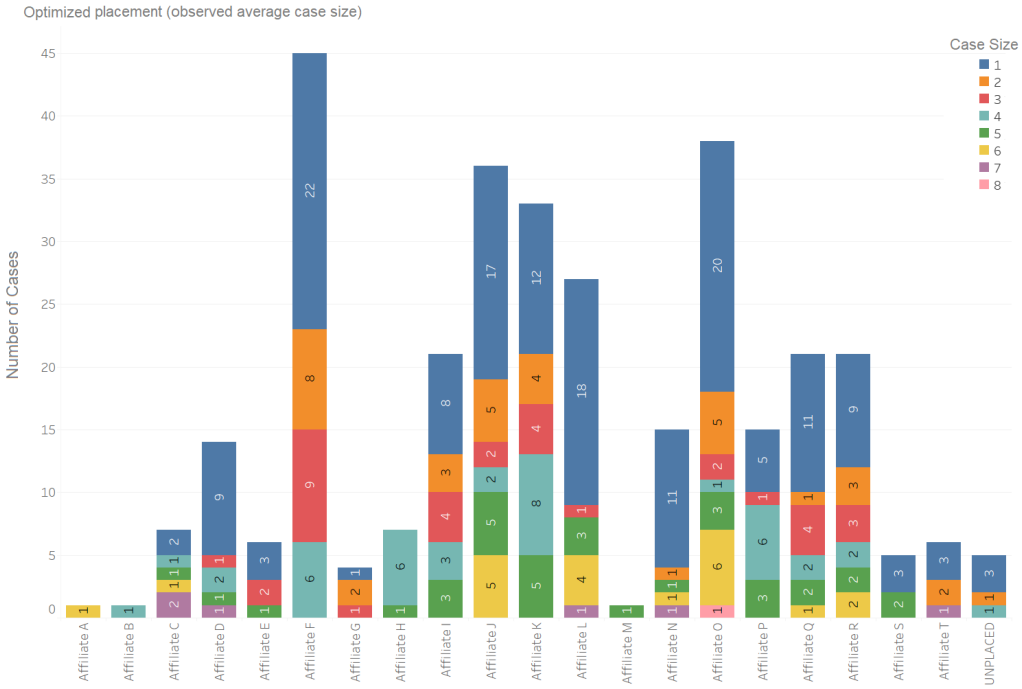


Figure 1.4: Average case size at each affiliate. Blue bar: observed average case size under HIAS placement. Orange bar: average case size for {observed capacity, activated binary service constraints, no minimum average case size} scenario. Red bar: average case size for {observed capacity, activated binary service constraints, at least observed average case size} scenario.



(a) Distribution of case sizes for {observed capacity, activated binary service constraints, no minimum average case size} scenario.



(b) Distribution of case sizes for {observed capacity, activated binary service constraints, at least observed average case size} scenario.

Figure 1.5: Distribution of case sizes at each affiliate.

1.6 Operationalizing Placement Software at US Resettlement Agency

Integer optimization and machine learning hold great promise of solving the operational challenge of improving placement outcomes in refugee resettlement. While these methods offer significant value, expertise is needed for successful implementation. In the private sector, this expertise is readily available. On the other hand, operations research in humanitarian environments, including refugee resettlement, typically feature significant challenges, such as lack of human and financial resources, lack of exposure to technology, and data scarcity. Humanitarian and non-profit organizations must be responsive to crisis events, immediate needs, and changes in political and donor climates. These realities can make it fairly prohibitive for resettlement agencies to be proactive in pursuing, and implementing, advanced technological innovations.

Successful integration of operations research methods in a humanitarian environment requires cultivating and sustaining partnerships with stakeholders that include both management, as well as practitioners that will use the technology. The authors of this paper worked closely with many dedicated members of staff at HIAS for many months to develop *Annie* into an innovative, interactive optimization environment for refugee resettlement. Our close working relationship built a level of rapport that allowed us to understand and remedy real operational challenges faced by resettlement staff. We believe these are key elements for creating a successful software solution for improving humanitarian operations.

1.6.1 Technologies Involved in the Creation of *Annie*

Annie represents the confluence of several open-source technologies, critical for this resource-constrained environment. In particular, the integer optimization problem RefMatch is modeled entirely within the PuLP Python modeling environment [41] and solved using the CBC [42] solver. The machine learning models described in Section 1.4 are based on the Python scikit-learn package. We chose to develop the interactive environment of *Annie* as a web application. The back-end is implemented in Python 3 using the Flask framework, with Jinja2 as the templating engine [43]. The front-end is a combination of HTML, CSS, and JavaScript. We made this choice of technology because it is modern, stable, accessible, and easy to build upon. The only installation that is needed is (the free) Python 3 and some freely available packages and libraries. Moreover, it is a light technology: The front-end operates entirely within a browser rather than as a downloadable, executable file. By combining core open-source integer optimization and machine learning technology within a flexible, modern interface, we were able to achieve a completely free, lightweight software solution for HIAS.

1.6.2 Interactive Optimization

Representing overall match quality in objective function (1.1a) is by no means trivial. Employment probabilities for refugees will always be estimated with error margins (see Appendix E for experiments and related discussions around uncertainty in employment probability estimation, and corresponding sensitivity of match outcomes). Even if the employment probabilities could be perfectly estimated, any algorithmic solution should be carefully evaluated before actual implementation, as the overall livelihoods of refugees are at stake. Therefore there is a

need for an interactive optimization environment, where resettlement staff can interact with various facets of the problem context. Without compromising on the insights afforded by the theory and data, *Annie* was designed to accommodate the real needs of the practitioner. The purpose of developing *Annie* as an interactive optimization tool is to translate advanced analytical methods into effective decision tools [see, e.g., 44]. The user of *Annie* is intimately involved in the matching process and can fine-tune the optimization results. We believe that *Annie* strikes the right balance. Our close interactions with HIAS allow us to iteratively develop and test multiple versions of the software via remote updating. Moreover, our predictive models can be refined as more data on 90-day employment outcomes arrive over time.

1.6.3 Features of *Annie*

Operationally, *Annie* optimizes for the expected number of employed *refugees* throughout the network of affiliates at HIAS. Alternative objective functions, such as those discussed in Appendix D, can be easily implemented.

The *Load Data* view is depicted in the rear left of Figure 1.6, where the optimization environment can be configured for the matching process, including the activation of binary support services. The matching results can be seen in the *View Results* view depicted in the front right of Figure 1.6, where the total expected number of employed refugees is prominently displayed near the top.

The output of the matching engine results in cases being optimally assigned to affiliates, depicted with user-friendly *tiles*. Figure 1.7 displays both case and affiliate tiles. Case tiles show language, nationality, and other attributes unique to the family, whereas affiliate tiles show support features offered by affiliates, along with the ability to quickly adjust capacity. Clicking on the tiles expands their size to reveal detailed information at a quick glance. Case tiles can be moved to other affiliates as desired. Figure 1.8 illustrates the ability to dynamically view changes in the match scores as refugee case tiles are moved from one affiliate to the

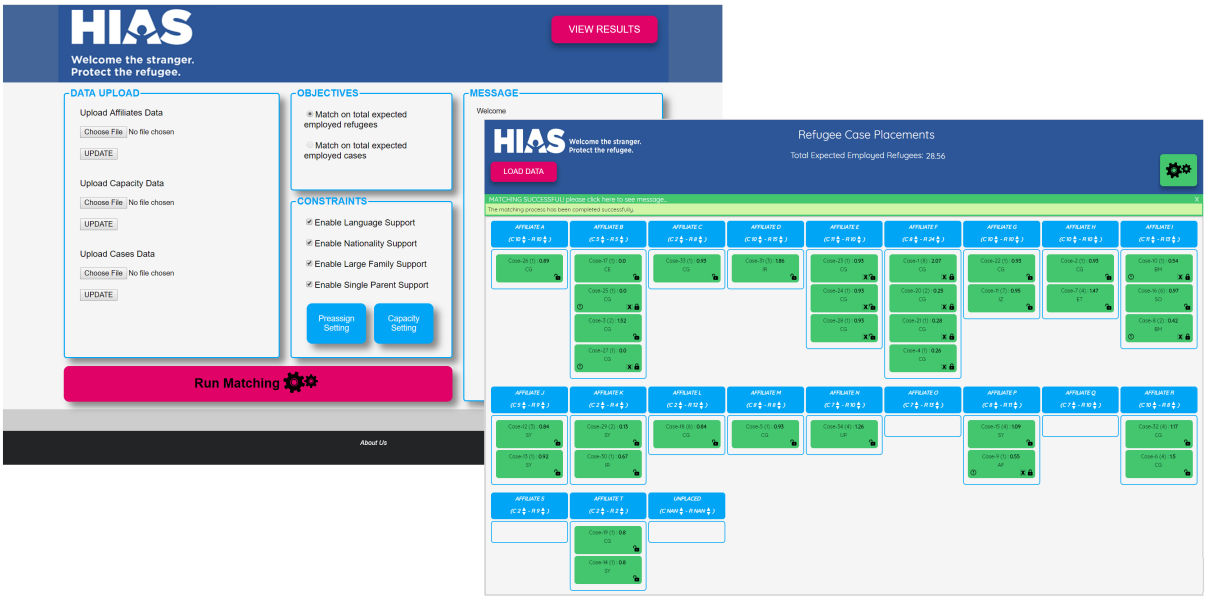


Figure 1.6: *Annie* Interface.

next. Moreover, the total expected number of employed refugees is also dynamically updated at the top of the *View Results* view. Hence, at a glance, the effect of moving cases to alternative affiliates is easily and clearly visualized.

Perhaps the most important feature of *Annie* is its ability for interactive optimization. Resettlement staff may interact with intermediate solver output in a manner that progresses toward eventual convergence of a finalized assignment of refugee cases to affiliates. This is facilitated through a lock icon on the case tile that resettlement staff can click, which locks desired case-affiliate matches. Figure 1.9 depicts this capability.

When locked, that case is temporarily "assigned" to that affiliate, and is literally unable to be moved elsewhere until unlocked. After locking certain case-affiliate matches (this essentially assigns $z_\ell^i = 1$ for family F^i and location L^ℓ), any remaining unlocked cases may be rematched, with affiliate capacities adjusted down from any locked cases, via a color-coded gray reoptimize button that indicates the non-optimized state (see Figure 1.9). Thus, any "final" matches can be locked, and all remaining cases can be rematched using the remaining available capacity.

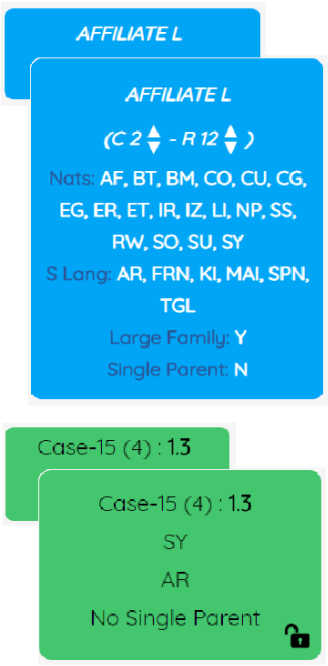
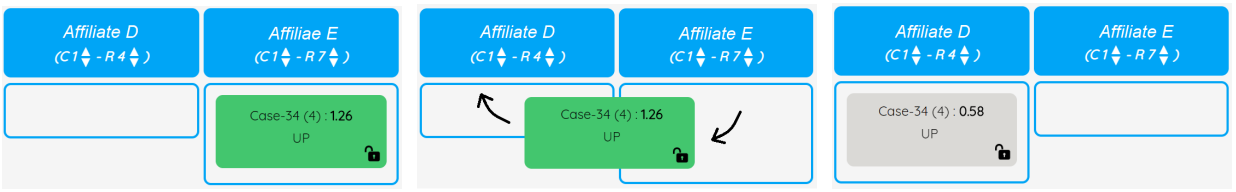


Figure 1.7: Expanding tiles: refugee and affiliate data.

We also enable cross-referencing. Cross-referencing occurs when refugee cases are linked to other cases that a) have previously been resettled to a specific local affiliate, or b) are among the pool of cases that are presently to be resettled to the same affiliate (note that these are cases with US ties, as previously described in Section 4). In either case, *Annie* visually depicts cases that are associated with a) an affiliate or b) other cases via unique yellow borders upon hovering over a large, boxed X icon, for associated case tiles. For any two cases i, i' that are cross-referenced, *Annie* sets $z_\ell^i = z_\ell^{i'}$ for all local communities L^ℓ; and if i, i' are cross-referenced to a particular local community ℓ', *Annie* sets $z_{\ell'}^i = z_{\ell'}^{i'}$ only for local community $L^{\ell'}$. Figure 1.10 depicts an example where two cases are cross-referenced not only to one another (e.g., adult siblings), but also to an affiliate.

If a case tile is moved into an affiliate but there is a lack of compatibility between this case and the new affiliate in terms of binary community support services, the background color of the case becomes red as an indication, and an exclamation mark icon appears in the bottom left of the case tile (see Figure 1.11). Hovering over this exclamation mark icon displays a new



(a) Case assigned to Affiliate E. (b) Moving case tile to Affiliate D. (c) Case tile moved to Affiliate D.

Figure 1.8: Case tiles can be moved by dragging to an alternate affiliate tile. Upon moving, the match scores dynamically update. The background of the case tile changes to gray to indicate a non-optimized state.

list that shows the unsupported needs for that particular case-community match.

Throughout the development process, we have firmly maintained that *Annie* is a tool that augments the perspective of resettlement staff at HIAS. That is, matches generated by *Annie* are suggestive in nature. HIAS has complete discretion to match and rematch cases according to their expert judgment. In this way, we allow for the best of both worlds: leveraging the strengths of modern computational technology—machine learning, integer optimization, and interactive visualization—while arming human decision-makers with all available information to facilitate the decision-making process.

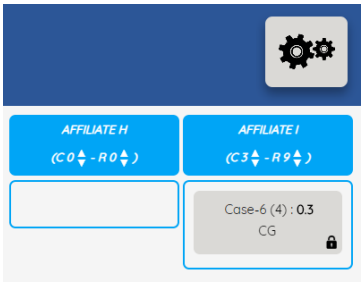


Figure 1.9: Locking case tiles and reoptimizing.

1.7 Conclusion

Refugee resettlement is a complex humanitarian challenge that requires insights from a number of disciplines, including operations research, statistics, economics, political science, and sociology. Much work is urgently needed to improve the livelihoods of resettled refugees and the communities into which they integrate. In this paper, we show how combining tools from machine learning, integer optimization, and interactive visualization can improve refugee outcomes within the US. We introduce the innovative software tool, *Annie* MOORE, that assists the US resettlement agency HIAS with matching refugees to their initial placements. Our software suggests optimal placements while giving substantial autonomy for the resettlement staff to fine-tune recommended matches. Because *Annie* matches on refugee employment outcomes, we expect refugees to more quickly integrate economically into each affiliate, as well as make more productive economic and societal contributions such as creating new jobs and generating tax revenues, benefitting local communities.

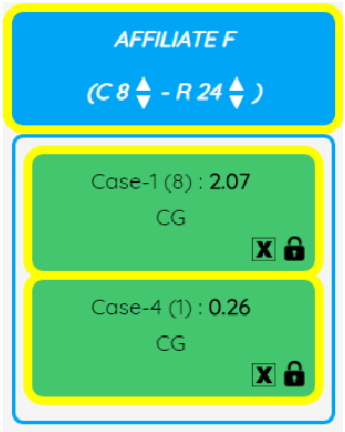


Figure 1.10: Cross-referencing cases to Affiliate F.

Annie has analytically enhanced the placement decision-making process at HIAS, having largely eliminated the inefficiencies of the former manual placement process. The operational process of placing refugees has improved considerably, enabling resettlement staff to place greater emphasis on cases that need greater attention, such as those with severe medical conditions.

Technological solutions, including machine learning and integer optimization, have enormous potential to help tackle humanitarian operations problems, such as placement optimization in refugee resettlement. While the humanitarian sector offers many opportunities for impact, any solution must properly account for the severe lack of resources—including financial, labor, time, and data. These factors must be carefully considered in designing solutions, to afford the best opportunity of effecting change. Particular solution design features that we advocate include being lightweight, open-source, and designed with the end-user in mind by incorporating important aspects of their regular operations.

There are several directions for further work. First, as is often the case in the humanitarian context, data has been difficult to obtain due to a severely resource-constrained environment. Indeed, data collection appears to be under-prioritized across the resettlement agencies. We used the only existing outcome data from previous US placements, namely a refugee-specific binary indicator for employment measured 90 days after arrival. While we went through great efforts to make the most out of the available data, the relative lack thereof necessarily hampered our prediction ability. Further work could apply our techniques to data on other outcomes, such as longer-term employment, physical and mental health, education, and household earnings. Unfortunately, at the time of writing, no data on these objectives for resettled refugees arriving in the US appears to be systematically available. However, we anticipate to be able to better process other constraints like free-form text fields to discern whether refugees require medical accommodations such as wheelchair access.

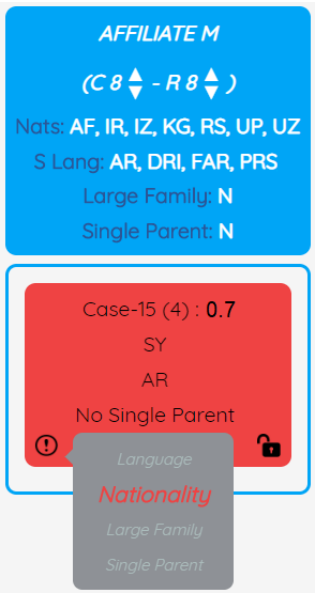


Figure 1.11: Case tile changes color when placed into affiliate that violates binary service constraints. Hovering over exclamation point reveals additional details.

Second, while annual approved arrival capacities exist for affiliates, refugees arrive stochastically over the course of a year. Therefore, it is important to schedule the arrival of refugees given the partial information about future arrival over the course of the whole year. [45] tackle this problem in the Swedish context.

Third, it is interesting to consider which features of local areas offer the best potential to host refugees. For example, we could analyze to what extent local unemployment or community demographics affect refugee outcomes. This could help refugee agencies target areas for new affiliates.

Fourth, the social objective considered in this paper is to maximize employment. Even if it can be argued that "there is no single, generally accepted definition, theory or model of immigrant and refugee integration" [46, p.114], it is also clear that there are key aspects of integration beyond employment. Ager and Strang [25], for example, argue that there are ten established integration indicators, including health, housing, and education. This additional information—such as housing information, social networks, or new job opportunities—likely exists to at least some degree at the local community level, and could prove very useful in supplementing the decision process. Moreover, regular and sustained engagement of local communities and associated stakeholders can also produce valuable insights that augment decision outcomes [see, e.g., 47]. So, while 90-day binary employment outcomes are at present the only data available to estimate future integration outcomes, additional integration factors may be possible to integrate in the future, and we thus leave the analysis of such models for future research.

Fifth, recent theoretical work on refugee matching [14, 16, 18, 48] suggests that preferences of refugees should be explicitly taken into account, because refugee families themselves know best where they would like to settle and where they are most likely to thrive. Refugee preferences could ideally be collected during the refugee pre-arrival orientation using a questionnaire that elicits how refugees might trade off features of areas (such as climate, urban

versus rural, crime, amenities, and quality of schools). Unfortunately, refugee preferences are not elicited either by UNHCR or by the US Department of State. In any case, the consideration of refugee preferences should be handled with care. Including preferences while optimizing for a particular observable outcome can in itself be problematic [49], and it is also unclear how preferences should be elicited based on the reported information. Allowing refugees to report complete preferences may also be overly challenging. Hence, although it is clear that the approach we adopt has room for improvement, we believe it to be a reasonable approach in line with the growing evidence that the initial placement of refugee families greatly affects lifetime employment which, in turn, profoundly alters lifetime welfare [7, 11, 12].

A final challenge with eliciting refugees' preferences—and a main theme in the book by Roth [50]—is that agents often find it "unsafe" to report true information. Rather than strategically misrepresenting information to "game" the system, agents may be reluctant to report complete information simply because a lack of knowledge on how the information will used, how it will be spread, or trust that the reported information will be used in their best interest. This is surprisingly often the case even in applications where the outcome has a large impact on future welfare and life quality, such as in school choice [51] and kidney exchange [52]. It is in general difficult to design systems where all agents find it "safe" to report true and complete preferences [see, e.g., 53–56]. Recent work on refugee matching with preferences also indicates that it can be difficult to design matching systems in which refugees have no incentive to misreport their preferences [e.g. 15, 18].

Annie has primarily been developed to assist HIAS in their initial refugee placements, and there is significant potential to expand to additional resettlement contexts, both within the US, as well as beyond—the most direct being with other US resettlement agencies which face analogous placement decision challenges. *Annie* could be used to help improve placement in the (Syrian) Vulnerable Persons Resettlement Scheme operated by the British government between 2015 and 2020. A recent report by the UK Independent Chief Inspector of Borders and Immigration recommended that the Home Office "improve the geographical matching process" of refugees in this resettlement scheme [57, p. 12]. In Sweden, asylum seekers who enter are temporarily placed at Migration Board accommodation facilities in anticipation of either a deportation order or a residence permit. If a residence permit is granted, the legal responsibility for asylum seekers (such as finding housing and schooling) is transferred from the Migration Board facility to one of the 290 municipalities in Sweden (43,745 such transfers were made in 2018). This system is, in a sense, a version of refugee resettlement in which asylum seekers are resettled within Sweden. While the current Swedish system is not based on sophisticated matching techniques, a recent report by the Swedish Government [58, p. 280] recommends that carefully designed optimization and matching systems should be adopted (indeed, *Annie* could be adapted for the Swedish context; the authors of this paper have already presented the first version of *Annie* at the Swedish Ministry of Finance for potential adoption). Finally *Annie* may, for example, be adapted for distributing asylum seekers who are currently at reception centers in host countries (such as Germany, or the southern border of the US).

Chapter 2

Dynamic Placement in Refugee Resettlement[1]

Employment outcomes of resettled refugees depend strongly on where they are placed inside the host country. While the United States sets refugee capacities for communities on an annual basis, refugees arrive and must be placed over the course of the year. We introduce a dynamic allocation system based on two-stage stochastic programming to improve employment outcomes. Our algorithm is able to achieve over 98 percent of the hindsight-optimal employment compared to under 90 percent of current greedy-like approaches. This dramatic improvement persists even when we incorporate a vast array of practical features of the refugee resettlement process including indivisible families, batching, and uncertainty with respect to the number of future arrivals. Our algorithm is now part of the *Annie*™ Moore optimization software used by a leading American refugee resettlement agency.

2.1 Introduction

There are 26 million refugees around the world [59]. The United Nations High Commissioner for Refugees (UNHCR) considers over 1.4 million to be in need of *resettlement*, that is, permanent relocation to a third country [22]. Resettlement is mainly targeted at the most vulnerable refugees, such as children at risk, survivors of violence and torture, and those with urgent medical needs. Dozens of countries around the world resettle refugees, but every year the number of refugees in need of resettlement far exceeds the number that is actually resettled. In 2019, around 63,000 refugees were resettled [22].

Historically, most hosting countries have paid little attention to which communities inside the country the refugees are resettled to. However, there is a great deal of evidence that the initial local resettlement destination dramatically affects the outcomes of refugees [7–12, 20, 60]. One specific variable impacted by community placement is whether and when resettled refugees find employment. Employment plays a key role in the successful integration of a

[1]Narges Ahani, Paul Gölz, Ariel D. Procaccia, Alexander Teytelboym, and Andrew C. Trapp. Dynamic Placement in Refugee Resettlement. *EC'21: The 22nd ACM Conference on Economics and Computation Proceedings.* 2021. https://arxiv.org/abs/2105.14388

refugee by "promoting economic independence, planning for the future, meeting members of the host society, providing opportunity to develop language skills, restoring self-esteem and encouraging self-reliance" [25].

As a result, in 2017, the American resettlement agency HIAS began to match refugees to communities by seeking to maximize the total number of refugees who obtain employment soon after arrival [61]. Refugees are allocated to HIAS over the course of the year, and HIAS must assign these refugees to a community immediately and irrevocably. However, HIAS's matching software *Annie*™MOORE (Matching and Outcome Optimization for Refugee Empowerment) allocates the weekly batches of arriving refugees in a greedy way. That is, each batch of arrivals is allocated by separately maximizing the expected employment of this batch, subject to not exceeding remaining community capacities and subject to ensuring that refugees have access to relevant services. By contrast, in order to optimize employment in such a *dynamic* setting, an algorithm must carefully manage the capacities by weighing in every placement decision whether a slot of capacity is more useful for placing the current refugee or a yet-unknown refugee arriving later in the year.

In this paper, we tackle the dynamics of refugee resettlement from an optimization viewpoint. We design two algorithms—one based on stochastic programming and another based on Walrasian equilibrium—that optimize the dynamic matching of refugees to communities in the United States. Our resettlement model is rich and captures all of the relevant practical features of the refugee resettlement process including indivisible families of refugees, batching, and unknown numbers of refugee arrivals. We evaluate the performance of our algorithms on HIAS data from 2014 until 2019. We show that both algorithms achieve over 98 percent of the hindsight-optimal employment in all years compared to a greedy (myopic) algorithm which typically achieves only around 90 percent. This is quite remarkable because the two algorithms we deploy are only known to have good theoretical properties in a rather unrealistic case of our model (with known arrival numbers and without families or batching). We then describe how we implemented our algorithms within *Annie*™.

2.1.1 Related Work

Several papers have studied theoretical models of refugee resettlement, which can be either static [14, 18, 19, 62] or dynamic [45, 63]. Inspired by wildly successful applications of matching theory, these papers aim to match refugees to communities based on the preferences of the refugees, of the communities, or of both, while satisfying constraints that are unique to the resettlement setting.

The data-driven approach that has been used for refugee resettlement in practice, however, is quite different. Introduced by Bansak et al. [20], the approach seeks to place refugees (subject to constraints) in a way that maximizes predicted employment. This consists of two components: using machine learning to estimate the probability that a given refugee placed at a given community would find employment, and using mathematical programming to perform the optimization. Ahani et al. [61] adopted a similar approach to develop *Annie*™; they also pointed out the practical relevance of indivisible families and the possibility of batching. Both papers seek to maximize employment with respect to a current batch of refugees, without considering future arrivals; it is in this sense that we think of deployed algorithms as *greedy*, and that is indeed our benchmark in this paper.

In very recent work that is independent from and concurrent to ours, Bansak [64] also

considered dynamic refugee resettlement; the algorithm obtaining the highest employment in that study is equivalent to our two-stage stochastic programming formulation in the simplest setting. Our model is much richer as we include non-unit family sizes, incompatibilities between families and communities, and allow for uncertain arrival numbers. Moreover, Bansak encountered computational difficulties that we are able to avoid by leveraging several algorithmic ideas. As a result, Bansak developed a large variety of algorithms that lead to lower employment in exchange for better running time. He evaluated his algorithms on a single month of arrivals; by contrast, our simulations are conducted over entire years, which becomes possible as our algorithms are many times faster.

On a technical level, our matching problem closely resembles the *online generalized assignment problem* [65] and its subproblem of *display ads*, which corresponds to matching individual refugees rather than indivisible families. Given that the resettlement agencies have access to data on past arrivals, refugee resettlement calls for algorithms that can capitalize on this data rather than for algorithms that are designed to be robust against worst-case distributions of refugees [65, 66]. To apply the most relevant algorithms [67, 68] we would have to assume that refugees can be partitioned into few *types*, which would determine employment probabilities. However, our employment predictions are based on more than 20 features, which makes this approach impractical in our setting. We also note that the idea of using shadow prices to guide an online matching was previously used by Devanur and Hayes [69] and by Vee et al. [70].

Further afield, our algorithms are related to previous work in kidney exchange. In particular, Dickerson et al. [71] and Dickerson and Sandholm [72] developed algorithms that seek to optimize the number of patients who received a kidney (or other objectives) despite uncertainty about arrivals and departures of patients and donors over time. On a high level, our technical approach is inspired by theirs in that in each stage we maximize immediate benefit minus a measure of "lost potential" for future benefit. To calculate this potential, however, the foregoing authors used automated parameter-tuning algorithms, which we discarded early on as they performed poorly in our experiments.

2.1.2 Organization of the Paper

This paper is organized as follows. In Section 2.2, we provide an overview of the US refugee resettlement process. In Section 2.3, we outline our model of dynamic refugee matching. In Section 2.4, we design one of our algorithms based on two-stage stochastic programming and show how it performs in a baseline setting that ignores the indivisibility of families, batching, and uncertainty of arrivals. In the next three sections, we layer on complexity toward our final setting: indivisible families (Section 2.5), batching (Section 2.6), and unknown arrival numbers (Section 2.7). We then explain how we implemented our approach within *Annie*™ in Section 2.8 and conclude in Section 2.9. In the appendix, we describe a second algorithm and provide further empirical results.

2.2 Institutional Background

The federal Office of Refugee Resettlement was created by the Refugees Act in 1980. The Act authorized the President of the United States to set annual capacities for resettlement and established funding rules. The resettlement process is managed by the US Refugee Admissions Program (USRAP) of the US Department of State, in conjunction with a number of federal

agencies across federal departments as well as the International Organization for Migration and the UNHCR.

Applications for the resettlement program take place from outside of the US, typically in refugee camps. The US government conducts security checks, medical screening, and performs cultural orientation, which can take upwards of 18 months [73]. After clearance, USRAP decentralizes the process of welcoming refugees to nine NGOs known as *resettlement agencies*, of which one is HIAS. Each agency works with their own network of *affiliates* supported by local offices, as well as religious entities like churches, synagogues or mosques, which serve as community liaisons for refugees. Each agency typically works with dozens of affiliates, though the number of affiliates can fluctuate over time. Some affiliates lack services to host certain kinds of refugees. For example, certain affiliates do not have translators for non-English-speaking refugees or lack support for single-parent families.

Agencies have no influence on what refugees are cleared for resettlement by USRAP or on when the refugees might arrive. Resettlement agencies meet on a weekly or fortnightly basis in order to allocate among themselves the refugees that have been cleared by USRAP. Refugees are usually resettled with members of their family. Such an indivisible group of refugees is referred to as a *case*. As a family can split when its members are fleeing their home country, some refugees who are applying for resettlement might already have existing relatives or connections in the US. Such cases *with US ties* are automatically resettled near their existing ties. All other refugees, referred to *free cases*, can be resettled by any agency into any of the agency's affiliates.

Each affiliate has an assigned annual capacity for the number of refugees (rather than cases) it can admit in a given fiscal year.[2] These capacities are approved by USRAP and, in theory, agencies cannot exceed them. In practice, capacities can be slightly adjusted towards the end of the year or, as in recent years, substantially revised in the course of the year. Since capacities limit the number of refugees *arriving* in a fiscal year rather than *allocated* in it, and since there is typically a delay of multiple months between the two events, the Department of State tells the resettlement agencies an estimated arrival date for each cleared case. Agencies are assessed annually by USRAP on their performance in finding employment for refugees within 90 days of their arrival. Data on 90-day employment is therefore diligently collected by the affiliates and monitored by the agencies.

2.3 Model

An instance of the matching problem first defines a set L of *affiliates*. Each affiliate ℓ has a capacity $c_\ell \in \mathbb{N}_{\geq 0} \cup \{\infty\}$ of how many refugees it can host. We call a collection $\{c_\ell\}_{\ell \in L}$ of capacities for all affiliates a *capacity profile* $\boldsymbol{c}$. To describe changes in capacity, it will be useful to manipulate the capacity profiles as vectors. Specifically, we write $\boldsymbol{c} - \boldsymbol{e}_\ell$ to describe the capacity profile obtained from $\boldsymbol{c}$ by reducing the capacity of affiliate ℓ by 1.

On the other side of the matching problem is a set $N = \{1, \ldots, n\}$ of *cases*. Each case i represents an indivisible family of $s_i \in \mathbb{N}_{\geq 1}$ refugees. Furthermore, each case i, for each affiliate ℓ, has an *employment score* $u_{i,\ell}$, which indicates the expected number of case members that will find employment if the case is allocated to ℓ. Typically, these employment scores $u_{i,\ell}$

[2]Each fiscal year ranges from October 1 of the previous calendar year to September 30. For example, fiscal year 2017 ranges from October 1, 2016 to September 30, 2017.

are real numbers in $[0, s_i]$, but we will also allow to set $u_{i,\ell} = -\infty$ to express that case i is not compatible with affiliate ℓ. We will refer to the combination of a case's size and vector of employment scores as the *characteristics* of the case. To ensure that the matching problem is always feasible, we will assume that L contains a special affiliate $\perp$ that represents leaving a case unmatched, where $u_{i,\perp} = 0$ for all cases i and $c_\perp = \infty$.[3]

We use the employment scores developed by Ahani et al. [61], and we give details on data preprocessing and training in Appendix G. Throughout this paper, we consider these employment scores as ground truth, which means that we evaluate algorithms directly based on the employment scores. An evaluation of how accurately the employment scores predict employment outcomes is outside of the scope of this paper, and we refer the reader to Ahani et al. for such an evaluation.

The goal of the matching problem is to allocate cases to affiliates such that the *total employment*, that is, the sum of employment scores, is maximized, subject to not exceeding capacities. For a set $I \subseteq N$ and a capacity profile $\boldsymbol{c} = \{c_\ell\}_{\ell \in L}$, define $\textsc{Matching}(I, \boldsymbol{c})$ as the matching integer linear program (ILP) below, where variables $x_{i,\ell}$ indicate whether case $i \in I$ is matched to affiliate $\ell \in L$:

$$\begin{aligned}
\text{maximize} \quad & \sum_{i \in I} \sum_{\ell \in L} u_{i,\ell}\, x_{i,\ell} \\
\text{subject to} \quad & \sum_{\ell \in L} x_{i,\ell} = 1 && \forall i \in I \\
& \sum_{i \in I} s_i\, x_{i,\ell} \leq c_\ell && \forall \ell \in L \\
& x_{i,\ell} \in \{0, 1\} && \forall i \in I, \ell \in L.
\end{aligned}$$

Let $\textsc{Opt}(I, \boldsymbol{c})$ denote the optimal objective value of $\textsc{Matching}(I, \boldsymbol{c})$. The *linear programming* (LP) *relaxation* of $\textsc{Matching}(I, \boldsymbol{c})$ is obtained by replacing the constraint $x_{i,\ell} \in \{0, 1\}$ by $0 \leq x_{i,\ell} \leq 1$ for all $i \in I, \ell \in L$. For a fixed matching, we define the *match score* of a case i as its employment score u_{i,ℓ_i} at the affiliate ℓ_i where it is allocated; we will also refer to its *match score per refugee*, $u_{i,\ell_i}/s_i$.

Finally, cases arrive *online*, that is, they arrive one by one and, when case i arrives, the decision of which affiliate to place it in must be made irrevocably, before the characteristics of the subsequent arrivals $i + 1, \ldots, n$ are known.[4] Thus, although an online matching algorithm must still produce a matching whose indicator variables $x_{i,\ell}$ satisfy the constraints of $\textsc{Matching}(N, \boldsymbol{c})$, the total employment $\sum_{i \in N, \ell \in L} u_{i,\ell}\, x_{i,\ell}$ typically will not attain the benchmark $\textsc{Opt}(N, \boldsymbol{c})$ of the optimal matching in hindsight. While we will not commit to a specific model of how the characteristics of arriving cases are generated, these arrivals should be thought of as *stochastic* rather than worst-case, and the distribution of case characteristics as changing slowly enough that sampling from recent arrivals is a reasonable proxy for the distribution of future arrivals.

[3]For example, allowing cases to be unmatched is necessary since an arriving case might only be compatible with affiliates whose capacity is already exhausted. When these situations occur in practice, such cases do not remain unmatched; instead, capacities can be increased or case–affiliate incompatibilities overruled manually by the arrivals officer. For our sequence of models, we report the fraction of matched refugees in Appendix I.0.4, and find that our algorithms do not lead to fewer refugees being matched than in the greedy baseline. To lower the number of unmatched refugees at the cost of reducing employment, one can add a constant reward per refugee to the $u_{i,\ell}$ with $\ell \neq \perp$.

[4]From Section 2.6 onward, cases will instead arrive in batches, which can be allocated simultaneously.

Throughout the following sections, we will consider a sequence of models, which incorporate an increasing number of features of the real-world refugee allocation problem: in Section 2.4, we consider traditional online bipartite matching, which results from requiring $s_i = 1$ in the above model; from Section 2.5 onward, we allow cases to have arbitrary size; from Section 2.6 onward, we also allow cases to arrive in batches rather than one by one; in Section 2.7, we no longer assume that the total number n of arriving cases is known to the algorithm.

2.4 Online Bipartite Matching ($s_i = 1$)

In this section, we will consider the special case of online bipartite (weighted) matching. We stress that this classical problem does not capture key features of the refugee-allocation problem in practice, which we will add in later sections. Instead, online bipartite matching serves as a starting point for the development of our matching algorithm since the simplified model allows us to justify our algorithmic approach by theoretical arguments. Later in the paper, we will empirically show that the approach continues to work well in richer and more realistic settings.

Formally, this section considers the model defined in the previous section, with the restriction that all cases consist of single refugees, i.e., that $s_i = 1$ for all $i \in N$. Under this assumption, it is well-known that the optimum matching for the ILP $\textsc{Matching}(I, \boldsymbol{c})$ can be found by solving its LP relaxation.

2.4.1 Algorithmic Approach

To motivate our algorithmic approach, we begin by describing why matching systems currently deployed in practice lead to suboptimal employment. These systems assign cases *greedily*, which—putting aside batching for now—means that an arriving case i is matched to the affiliate ℓ with highest employment score $u_{i,\ell}$ among those that have at least s_i remaining capacity. The main problem with greedy assignment is that it exhausts the capacity of the most desirable affiliates too early. In particular, we observe on the real data that a large fraction of cases have their highest employment score in the same affiliate ℓ^*, but that the size of the employment advantage of affiliate ℓ^* over the second-best affiliate varies. Since it only considers the highest-employment affiliate for each case, greedy assignment will fill the entire capacity of ℓ^* early in the year, including with some cases that benefit little from this assignment. Consequently, cases that would particularly profit from being placed in ℓ^* but arrive later in the year no longer fit within the capacity.

Intuitively, the decision to match a case i to an affiliate ℓ has two effects: the immediate increase of the total employment by $u_{i,\ell}$ but also an opportunity cost for consuming ℓ's capacity, which might prevent profitable assignments for later arrivals. Since greedy assignment only considers the former effect, it leaves employment on the table.

A better approach would be *two-stage stochastic programming*, which allocates an arriving case i to the affiliate ℓ maximizing the sum of the immediate employment $u_{i,\ell}$ and the expected optimal employment obtainable by matching the future arrivals subject to the remaining capacity. That is, if, at the time of i's arrival, the remaining capacities are given by $\boldsymbol{c}$, two-stage stochastic programming allocates i to the affiliate

$$\operatorname*{arg\,max}_{\ell \in L : c_\ell \geq s_i} \quad u_{i,\ell} + \mathbb{E}\Big[\textsc{Opt}\big(\{i+1, \ldots, n\}, \boldsymbol{c} - s_i \cdot \boldsymbol{e}_\ell\big)\Big],$$

where the expectation is taken over the characteristics of cases $j = i+1, \ldots, n$. Since adding a constant term does not change the argmax, this can be rewritten as

$$= \underset{\ell \in L: c_\ell \geq s_i}{\arg\max} \quad u_{i,\ell} - \mathbb{E}\Big[\textsc{Opt}\big(\{i+1,\ldots,n\}, \boldsymbol{c}\big)\Big] + \mathbb{E}\Big[\textsc{Opt}\big(\{i+1,\ldots,n\}, \boldsymbol{c} - s_i \cdot \boldsymbol{e}_\ell\big)\Big]$$

$$= \underset{\ell \in L: c_\ell \geq s_i}{\arg\max} \quad u_{i,\ell} - \mathbb{E}\Big[\textsc{Opt}\big(\{i+1,\ldots,n\}, \boldsymbol{c}\big) - \textsc{Opt}\big(\{i+1,\ldots,n\}, \boldsymbol{c} - s_i \cdot \boldsymbol{e}_\ell\big)\Big]. \tag{2.1}$$

Using our assumption that $s_i = 1$, this can be simplified to

$$= \underset{\ell \in L: c_\ell \geq 1}{\arg\max} \quad u_{i,\ell} - \mathbb{E}\Big[\textsc{Opt}\big(\{i+1,\ldots,n\}, \boldsymbol{c}\big) - \textsc{Opt}\big(\{i+1,\ldots,n\}, \boldsymbol{c} - \boldsymbol{e}_\ell\big)\Big].$$

Note that the expectation that is subtracted in either of the last two lines is exactly the expected opportunity cost of reducing the capacity of ℓ by placing case i there. This motivates our algorithmic approach: in every time step, we first compute a *potential* p_ℓ for each affiliate ℓ. Then, rather than myopically maximizing the utility of the match as does greedy assignment, our algorithm PM ("**p**otential **m**atch") myopically maximizes the utility of the current match minus the potential of the capacity used, as shown in Algorithm 1. (Note that an affiliate ℓ can always be defined in Line 5 as, by assumption, $c_\perp = \infty$.)

ALGORITHM 1: PM(Potential)

Parameter: a subroutine Potential to determine affiliate potentials

1 initialize the capacities c_ℓ for each affiliate ℓ;
2 **for** $t = 1, \ldots, n$ **do**
3 observe the case size s_t and the employment scores $\{u_{t,\ell}\}_\ell$;
4 call Potential() to define a potential p_ℓ for each affiliate ℓ;
5 $\ell \leftarrow \arg\max_{\ell \in L: c_\ell \geq s_t} u_{t,\ell} - s_t\, p_\ell$;
6 allocate case t to ℓ and set $c_\ell \leftarrow c_\ell - s_t$;

We estimate the expected value of the opportunity cost by averaging over a fixed number k of *trajectories*, each of which consists of randomly sampled characteristics of all arrivals $i+1$ through n. Since the characteristics of arriving refugees change over time, but since these changes tend to be gradual, we draw these arrival characteristics uniformly with replacement from the arrivals in the six months prior to the current allocation decision.

For each sampled trajectory, it remains to calculate the potential, which we would like to equal the opportunity cost $\textsc{Opt}\big(\{i+1,\ldots,n\}, \boldsymbol{c}\big) - \textsc{Opt}\big(\{i+1,\ldots,n\}, \boldsymbol{c} - \boldsymbol{e}_\ell\big)$. Clearly, this could be computed by solving $\mathcal{O}(|L|)$ matching problems. More easily still, celebrated results in matching theory [74] show that the opportunity costs for *all* affiliates with remaining capacity can be computed as the shadow prices of a *single* LP:

Fact 1 *Fix a matching-problem instance, in which all cases i have size $s_i = 1$. In the LP relaxation of MATCHING$(N, \boldsymbol{c})$, let $\{p_\ell\}_{\ell \in L}$ denote the unique element-wise maximal set of shadow prices for the constraints $\sum_{i \in N} s_i\, x_{i,\ell} \leq c_\ell$. Then, for each ℓ with $c_\ell \geq 1$,*

$$p_\ell = \textit{OPT}\big(\{i+1,\ldots,n\}, \boldsymbol{c}\big) - \textit{OPT}\big(\{i+1,\ldots,n\}, \boldsymbol{c} - \boldsymbol{e}_\ell\big).$$

This suggests the procedure Pot1 for computing potentials, which is shown in Algorithm 2. We also develop a second method Pot2 for computing potentials, based on a slightly different matching LP. This second methodology is motivated by a different theoretical argument, based on competitive equilibria rather than stochastic programming. For conciseness, we give this justification in Appendix H.

ALGORITHM 2: Pot1(k)

Parameter: $k \in \mathbb{N}_{\geq 1}$, the number of trajectories per potential computation

Input: remaining capacities $\boldsymbol{c}$, the index t of the last observed case, characteristics of cases arriving in the past 6 months

Output: a set of potentials p_ℓ for all affiliates ℓ

for $j = 1, \ldots, k$ **do**

  for each $i = t+1, \ldots, n$, set s_i and $\{u_{i,\ell}\}_\ell$ to the size and employment scores of a random, recently arrived case;

  solve the following bipartite-matching LP:

$$
\begin{aligned}
\text{maximize} \quad & \sum_{i=t+1}^{n} \sum_{\ell \in L} u_{i,\ell}\, x_{i,\ell} & & \\
\text{subject to} \quad & \sum_{\ell \in L} x_{i,\ell} = 1 & & \forall i = (t+1), \ldots, n \\
& \sum_{i=t+1}^{n} s_i\, x_{i,\ell} \leq c_\ell & & \forall \ell \in L \quad (*) \\
& 0 \leq x_{i,\ell} & & \forall i = (t+1), \ldots, n, \forall \ell \in L.
\end{aligned}
$$

  for each ℓ, set p_ℓ^j to the maximal shadow price[5] of the constraint $(*)$ for ℓ;

set $p_\ell \leftarrow (\sum_{j=1}^{k} p_\ell^j)/k$ for all ℓ;

return $\{p_\ell\}_{\ell \in L}$;

[5] One way of finding the maximal shadow price is to first solve the dual LP to find its objective value, then adding a constraint that constrains the objective of the dual LP to be equal to this optimal objective value, and to finally maximize the sum of dual variables p_ℓ over this new restricted LP.

2.4.2 Empirical Evaluation

We evaluate the employment of our potential-based matching algorithm on real yearly arrivals at HIAS. For each fiscal year, we consider all refugees who arrived in this period, and we consider them in the order in which they were received for allocation by HIAS. For the capacities, we use the year's *final*, i.e., most revised, capacities.[6] We also immediately take into account that affiliates are restricted in which nationalities, languages, and family sizes they can accommodate, as well as in whether they can host single parents.

The main way in which we deviate from reality in this experiment is accommodates the assumption in this section that cases have unit size. To satisfy this assumption, we split each case of size $s_i > 1$ into s_i identical single-refugee cases with a $1/s_i$ fraction of the original employment scores. In subsequent sections, we will repeat the experiments without this modification.

We study 6 fiscal years, from 2014 to 2019. As affiliates closed and opened across these years, the number of affiliates varies between 16 and 24 (not counting the unmatched affiliate $\perp$). Finally, the number of arriving refugees (cases) varies between 1,670 (640) and 4,150 (1,630) across fiscal years.

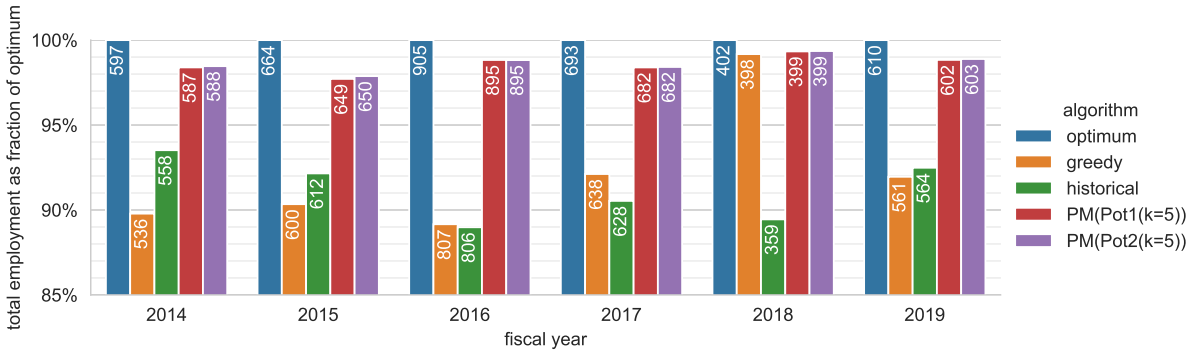


Figure 2.1: Total employment obtained by different algorithms, assuming that cases are split into multiple cases of size 1. Capacities are the final capacities of the fiscal year. For the potential algorithms, total employment is averaged over 10 random runs. The numbers in the bars denote the absolute total employment; the bar height indicates the proportion of the optimum total employment in hindsight.

As shown in fig. 2.1, even the greedy baseline obtains a total employment of between 89% and 92% of $\textsc{Opt}(N, \boldsymbol{c})$, the optimum matching in hindsight. (One outlier is the year 2018, which we discuss below.) Nevertheless, the greedy algorithm leads to between 50 and 100 fewer refugees finding employment every year compared to what would have been possible in the optimum matching. Our potential algorithms close a large fraction of this gap, obtaining between 98% and 99% of the optimal total employment. This holds both for algorithms based on Pot1 and for those based on Pot2. Since experiments in this model take much longer to run than those in subsequent models, we defer a comparison between the two potential methods and between values of k to section 2.6.1, where we can run the potential algorithms a sufficient number of times to discern smaller differences.

[6]When the number of refugees resettled in the fiscal year exceeds the official capacity, we use the the number of resettled refugees instead. In these situations, HIAS negotiated an increase in capacity that is not always recorded in our data.

The fiscal year 2018 stands out from the others due to the fact that the greedy algorithm performs on par with the potential algorithms, at 99% of the hindsight-optimal total employment. This is easily explained by the fact that the capacities are much looser than in other fiscal years: whereas, in all other fiscal years between 2014 and 2019, the number of arriving refugees amounts to between 84% (2019) and 97% (2016) of the final total capacity across all affiliates, this proportion is only 48% in 2018. Since capacity is so abundant, the optimal matching will match a large fraction of cases to their maximum-score affiliate, and the greedy matching is close to optimal.

We also compare to the employment obtained by the allocation chosen by HIAS ("historical"). This comparison gives the historical matching a slight advantage, as HIAS sometimes overrides the incompatibility between an affiliate and a case, which we do not allow any other algorithm to do.[7]

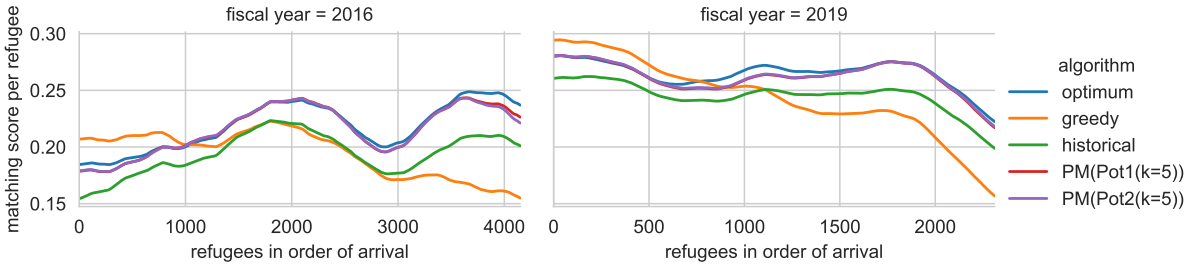


Figure 2.2: Evolution of the per-refugee match score in order of arrival, for fiscal years 2016 and 2019 in the experiment of fig. 2.1 (split cases, final capacities). Consecutive match scores are smoothed using triangle smoothing with width 500.

In fig. 2.2, we investigate how the match score changes over the course of two fiscal years, 2016 and 2019, chosen to contain a year in which the greedy and historical baselines perform relatively poorly and one in which they perform well. As the match score of subsequently arriving refugees can be very different, these graphs are heavily smoothed over time. If arrivals were independently drawn from a time-invariant distribution, we would expect the curve resulting from the hindsight optimum to be approximately constant for each fiscal year, since how much employment the optimum matching can extract from a case would be independent of the case's arrival time. This is not quite so, especially among the early refugees in fiscal year 2016 and the late refugees in fiscal year 2019, which seem to have worse employment prospects than other refugees in the plot. For most of each year, however, the curve of the hindsight optimum moves within a relatively narrow band.

The curves of both potential algorithms are nearly indistinguishable from one another, which shows that the algorithms make very similar decisions. In 2016, these curves start out closely tracking the curve of the optimal-hindsight matching, but fall behind for the last cases of the fiscal year, which we observe in most fiscal years. The similarity of the curves over most of the year indicates that our approach of sampling trajectories from past arrivals is nearly as useful as the optimum algorithm's perfect knowledge of future arrivals and that it leads to a similar trade-off in extracting immediate employment versus preserving capacity for later arrivals. Of course, the imperfect knowledge of the future incurs a small loss towards

[7] In these cases, we estimate the employment achieved by the case using the regression rather than using $u_{i,\ell} = -\infty$.

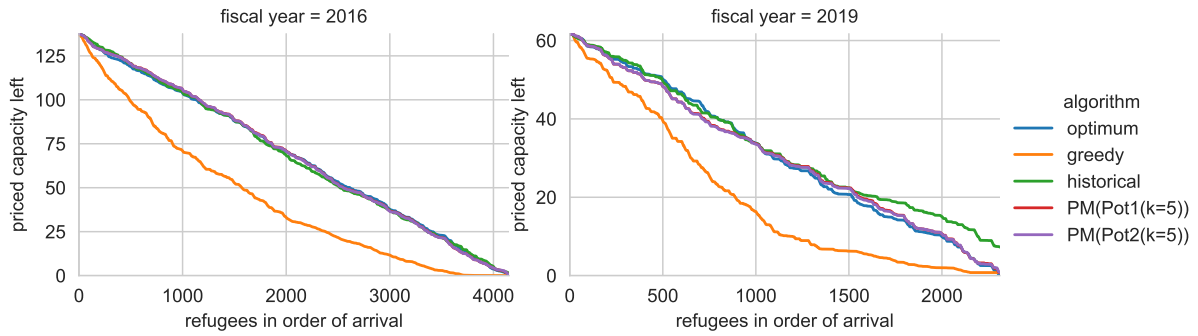


Figure 2.3: Remaining priced capacity at the time of arrival of different refugees, for fiscal years 2016 and 2019 in the experiment of fig. 2.1 (split cases, final capacities).

the end of the fiscal year, likely because the amount of capacity reserved per affiliate does not perfectly match the demand, which explains the gap in total employment between the hindsight optimum and the potential algorithms. This typical end-of-year effect is not very pronounced in fiscal year 2019, likely because the final arrivals of fiscal year 2019 have lower employment probabilities than what would be expected based on past arrivals. Instead, the potential algorithms fall behind the optimum algorithm for some period in the middle of the year, perhaps because they are reserving capacity for late arrivals which the optimum already knows to hold little promise.

The most striking curve is that of the greedy algorithm, which lies above those of all other algorithms in the first quarter of arrivals, but then falls clearly below the other curves in the second half. This observation can be explained by the effect we predicted in the motivation of our potential approach: the greedy algorithm extracts small additional gains in employment early in the arrival period, at the cost of prematurely consuming the capacity of the most desirable affiliates. Then, the lack of capacity limits the match scores of later arrivals, resulting in an overall unfavorable trade-off. This effect can be directly seen in fig. 2.3, in which we visualize the amount of capacity remaining in the most valuable affiliates. Specifically, looking at all arrivals of the fiscal year, we compute the shadow prices of the matching LP. At any point in time, we can then weight the remaining capacity by these prices to obtain a *priced capacity*. In fig. 2.3, we see that the optimum-hindsight matching and the potential algorithms use up the priced capacity at a roughly constant pace and essentially consume it all. By contrast, the greedy algorithm uses up the capacity very quickly, such that at the median refugee, only 22% (2016) or 17% (2019) of the priced capacity is left.

The historical matching made by HIAS does not have such obvious defects, but still falls short in terms of total employment. In both reference years, the average employment moves in parallel with the optimum matching, meaning that HIAS does not overly focus on extracting employment at certain parts of the fiscal year at the expense of others. However, the average employment consistently lies below that of the optimum and of the potential algorithms. We see in fig. 2.3 that, in 2019, HIAS started consuming the priced capacity at a near-constant pace very similar to that of the optimum algorithm. Around the median arrival, however, the historical matching slowed down its capacity consumption and ended up not consuming all priced capacity, which explains some loss in total employment. One reason for this behavior might be that HIAS staff treat the last 9% of the capacity as a reserve that they are more reluctant to use. In a year such as 2019, in which the overall arrivals were only 84% of the total

capacity, this heuristic might have actually kept much of the reserve capacity free, including in the affiliates that could have generated higher employment. By contrast, the total arrivals in 2016 amounted to 97% of the overall capacity, which could explain why nearly all priced capacity was consumed in this year. The fact that, in 2016, the historical assignment achieved lower average employment than the potential approaches despite using up priced capacity in a similar pattern indicates that whom this capacity is allocated to throughout the year was somewhat inefficient in terms of employment.

2.5 Non-Unit Cases ($s_i \geq 1$)

The most pressing aspect of refugee matching that we have ignored thus far is that many cases do not consist of individual refugees. Instead, they consist of an entire family of refugees, which has to be resettled to the same affiliate.

To accommodate cases consisting of multiple family members, we will from now drop the assumption that the s_i are 1. The main effect of this change is that the LP relaxation of the ILPs $\textsc{Matching}(I, \boldsymbol{c})$ can now be a proper relaxation. Indeed, the LP relaxation might allow for higher objective values because it allows fractional solutions.[8] As a result, our dual prices will no longer *exactly* compute the marginal value of a unit of capacity. In any case, to retain the exact connection to stochastic programming in eq. (2.1), PM would have to subtract the opportunity cost of s_i units of capacity from $u_{i,\ell}$, which might exceed s_i times the opportunity cost of a single unit of capacity.

However, as the capacity of most affiliates is much larger than the size of a typical case, both approximations can be expected to be relatively close, which is what we find empirically: we repeat the experiment of the previous section, but without splitting up cases into individual refugees. The results are nearly indistinguishable, which supports our decision to use LP relaxations even in the setting with indivisible cases. The full figures are deferred to appendix I.0.1.

2.6 Batching

A second aspect that we have not considered thus far is that HIAS does not actually process arriving cases one by one, but in batches containing one or multiple cases. Most of these batches result from the weekly meetings between the resettlement agencies, but smaller batches with urgent cases are allocated between the weekly meetings.

The fact that cases arrive in batches does not make the problem harder; after all, a matching algorithm that does not support batching can still be used by presenting the cases of each batch to the algorithm one by one. As we will argue, however, batching represents an opportunity to improve on this strategy: there is a (limited) opportunity to increase total employment and a (significant) opportunity to reduce running time.

Concerning total employment, using a non-batching algorithm in a batching setting is wasteful since it ignores potentially valuable information. Specifically, when the earliest cases of the batch are allocated, a non-batching algorithm presumes that the characteristics of the other cases in the batch are not yet known. Arguably, as the sizes of batches tend to be much

[8] One can always find a fractional solution that splits cases into $1/s_i$ fractions similarly to what we did in the evaluation of section 2.4.2.

smaller than the total number of cases n, the amount by which this can increase total employment is likely to be limited.

As for running time, given that the matching algorithm receives no new information between the first and last case of a batch, it seems reasonable not to recompute potentials within a batch. As there tend to be 5 to 10 times more cases than batches and as the computation of potentials is the bottleneck in the running time of the potential algorithms, this promises to substantially speed up the algorithm.

In adapting our algorithm PM to batching, we will not change how we compute the potentials p_ℓ. However, the algorithm now allocates all cases in the batch at once, still with the objective of optimizing the immediate utility of the assignment less the sum of potentials consumed. Thus, our extended algorithm PMB ("**p**otential **m**atch with **b**atching", algorithm 3) allocates the current batch according to the solution to a matching ILP, in which the utility of matching case i to affiliate ℓ is set to $u_{i,\ell} - s_i\, p_\ell$. Note that, if all batches have size $b = 1$, this algorithms coincides with our previous algorithm PM. Moreover, PMB also generalizes the greedy algorithm previously implemented in *Annie*™, which can be recovered by setting all potentials p_ℓ to zero.

ALGORITHM 3: PMB(Potential)

Parameter: a subroutine Potential to determine affiliate potentials

initialize the capacities c_ℓ for each affiliate ℓ;

$t_{last} \leftarrow 0$; // index of last case in previous batch

while $t_{last} < n$ **do**

  observe the size b of the current batch;

  $t \leftarrow t_{last} + b$; // index of last case in current batch

  observe the case size s_i and the employment scores $\{u_{i,\ell}\}_\ell$ for all $i = (t_{last}+1), \ldots, t$;

  call Potential() to define a potential p_ℓ for each affiliate ℓ;

  let $\{\hat{x}_{i,\ell}\}$ be an optimal solution to the following bipartite-matching ILP with knapsack constraints:

$$
\begin{aligned}
\text{maximize} \quad & \sum_{i=t_{last}+1}^{t} \sum_{\ell \in L} (u_{i,\ell} - s_i \cdot p_\ell)\, x_{i,\ell} \\
\text{subject to} \quad & \sum_{\ell \in L} x_{i,\ell} = 1 && \forall i = (t_{last}+1), \ldots, t \\
& \sum_{i=t_{last}+1}^{t} s_i\, x_{i,\ell} \le c_\ell && \forall \ell \in L \\
& x_{i,\ell} \in \{0,1\} && \forall i = (t_{last}+1), \ldots, t, \forall \ell \in L.
\end{aligned}
$$

  for $i = t_{last}+1, \ldots, t$ **do**

    allocate case i to unique affiliate ℓ where $\hat{x}_{i,\ell} = 1$;

    $c_\ell \leftarrow c_\ell - s_i$;

  $t_{last} \leftarrow t$;

2.6.1 Empirical Evaluation

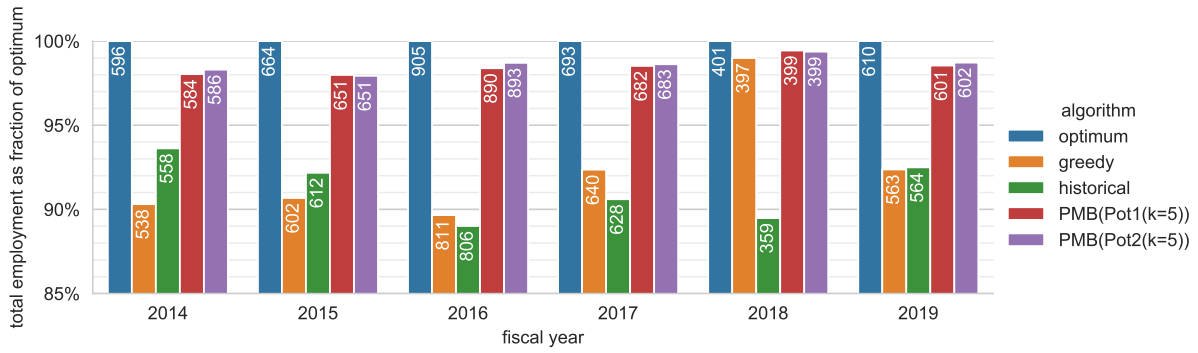


Figure 2.4: Total employment, where cases are not split and arrive in batches. Capacities are the final fiscal year capacities. In contrast to fig. 2.1, cases arrive in batches and the batching variants of greedy and the potential algorithms are used. For the potential algorithms, the mean employment across 50 random runs is shown.

We repeat the experiment measuring the total employment obtained by the algorithms, this time with the greedy algorithm and the potential algorithms allocating cases in batches. As shown in fig. 2.4, the results again look very close to those in the restricted setting of online bipartite matching, confirming that our algorithmic approach generalizes well not only to non-unit case sizes but also to batching as it is used in practice.

Since processing entire cases in batches is much faster than processing cases (or individual refugees) one by one, we are now in a position to run each potential algorithm many times and analyze the distribution of total employments. As shown in fig. 2.5, the total employment produced by each potential algorithm is sharply concentrated, especially for $k \geq 3$.

Running each algorithm many times enables us to compare the relative performance of the potential algorithms. Across both ways of computing potentials, and all fiscal years (with the exception of 2018, where everything is very close together), we see a clear tendency that averaging the potentials across more trajectories improves the employment outcome. These effects are somewhat limited, though, as going from a single trajectory to nine trajectories improves the median employment by less than half a percent of the hindsight optimum. As is to be expected, there appears to be diminishing returns in increasing k.

For k held constant, we observe that the Pot2 variants quite consistently outperform the Pot1 variants; again with the exception of 2018, in which a small inversion of this trend can be seen. While all potential algorithms perform very well, based on these results, we recommend the Pot2 potentials with a relatively large k for practical implementation. Of course, increasing k increases the running time of the matching algorithm. However, since a resettlement agency computes only one set of potentials per day, the algorithm runs in few seconds even for $k = 9$ (see appendix I.0.2).

2.7 Uncertainty in the Number of Future Arrivals

Given that our algorithm PMB supports non-unit sized cases and batching, it might seem that we are ready to replace the greedy algorithm in *Annie*™ by our potential algorithm. However,

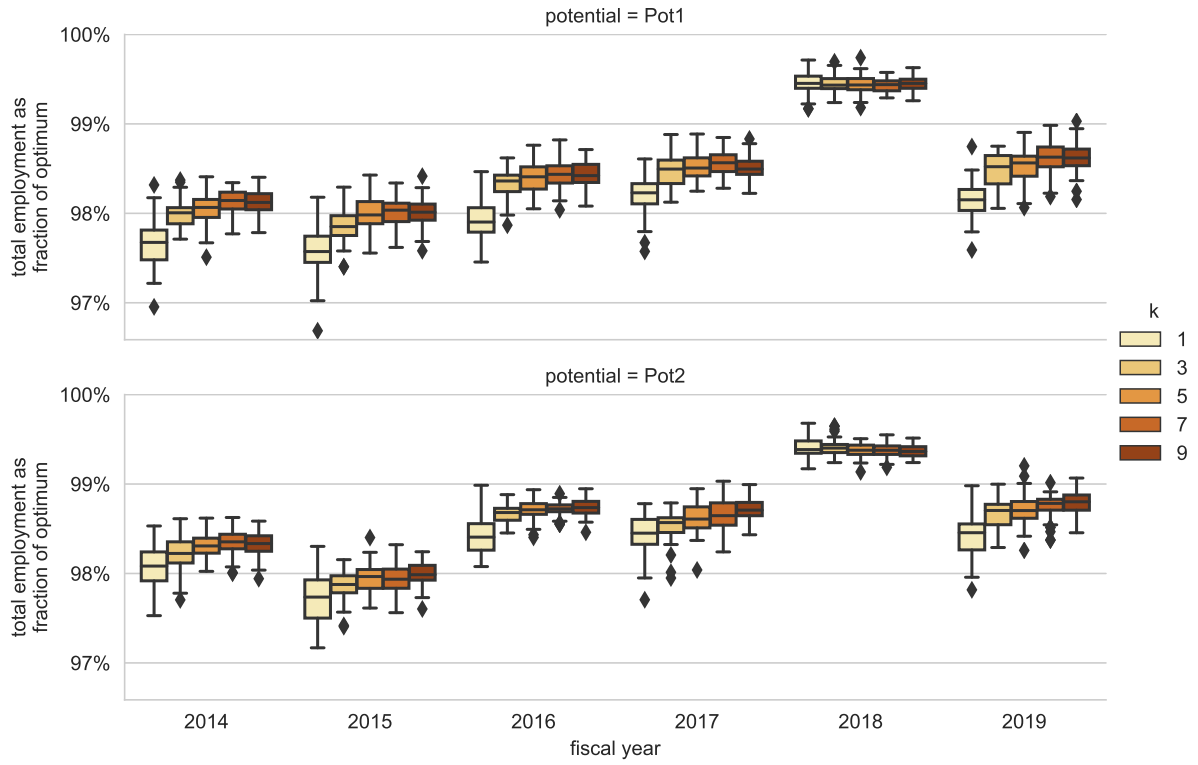


Figure 2.5: Distribution of the total employment obtained by instantiating PMB with different potential methods and different k, in the experiment of fig. 2.4 (whole cases, batching, final capacities) and over 50 random runs per algorithm.

our algorithm crucially relies on one piece of input that the greedy algorithm did not need, namely, the total number of cases arriving in the fiscal year. This number determines the length of the sampled trajectories, which can greatly impact the shadow prices and, thus, how the algorithm allocates cases.

In principle, the information given to resettlement agencies should provide a fairly precise estimate of how many cases are expected to arrive. Indeed, before the start of each fiscal year, the US Department of State announces how many refugees it intends to resettle in that fiscal year, and resettlement agencies are instructed to prepare for a certain fraction of this total number. In fact, HIAS sets its affiliate capacities to sum up to 110% of this number of announced refugees, which is intended to give local affiliates a good idea of how many refugees they will receive while affording the resettlement agency some freedom in its allocation decisions.

2.7.1 Relying on Capacities

It is thus natural to run our potential algorithms under the assumption that the number of arriving refugees will be $1/(110\%) \approx 91\%$ of the total announced capacity.[9] The result of this

[9]To convert the number of remaining refugees into a number of cases, we divide by the average case size of recent arrivals (over the years, this average size fluctuates between 2.4 and 2.6). While the number of refugees who have arrived is below 91% of the total capacity, this gives us a total number of cases n for the algorithms. Once the number of arrivals exceeds 91% of the total capacity, we make the algorithms assume that the current case is the last to arrive, that is, all subsequently sampled trajectories have length zero.

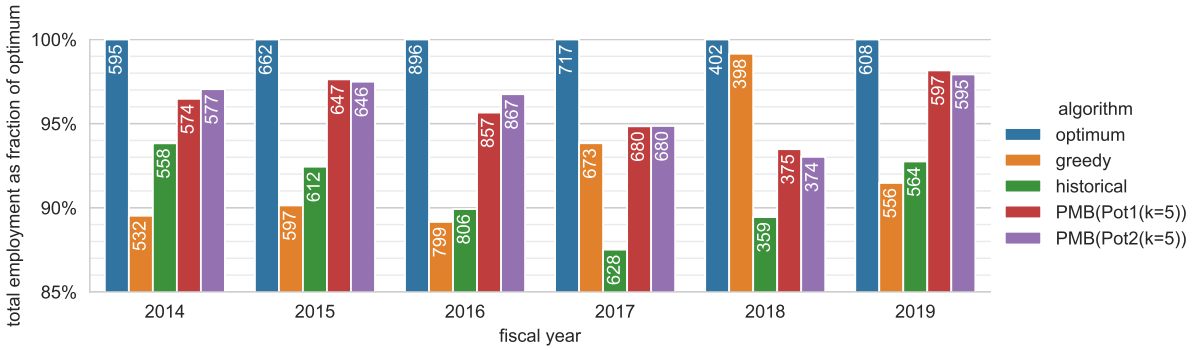


Figure 2.6: Total employment, where cases are not split up and arrive in batches. The potential algorithms no longer have access to the true number of arriving cases but assume that the arriving refugees amount to 91% of the total capacity. Capacities are the *initial* capacities of the fiscal year (except for historical). For the potential algorithms, the mean employment across 50 random runs is shown.

strategy is shown in fig. 2.6. As we choose the initial, unrevised capacities, the employment scores of the hindsight optimum and the greedy algorithm may differ from those in previous experiments, which used the most revised capacities.[10] In all fiscal years other than 2017 and 2018, the imprecise knowledge of future arrivals deteriorates the approximation ratio of the potential algorithms, but the potential algorithms continue to clearly outperform the greedy baseline, and they outperform the historical matching on all years.

Putting aside the years of 2017 and 2018 for now, we investigate the fiscal years 2016 and 2019, in which arrivals were highest and lowest relative to the announced capacity. In fiscal year 2016, the total arrivals were particularly large relative to the initial capacity: the arrival numbers added up to 100% of the initial capacity rather than 91%, which means that our potential algorithms expected around 3,770 refugees to arrive rather than the 4,150 that ended up arriving. As a result, the potential algorithms consume the priced capacity at an approximately constant rate, consuming it all around the expected number of expected refugees (fig. 2.7, bottom left). Up to this point, the potential algorithms are more generous in consuming capacity than would be ideal given the actual number of arriving cases, which is why the potential algorithms obtain a slightly higher average employment over the first three quarters of arrivals (fig. 2.7, top left) than the optimal matching in hindsight. For refugees arriving after the 3,770 expected refugees, however, the capacity in the best affiliates is used up, which is why the averaged employment sharply drops after this point.[11]

In 2019, by contrast, fewer refugees arrived than expected, only 86% of the total capacity. At the bottom right of fig. 2.7, it is visible that the potential algorithms consume priced capacity at a slightly lower rate than the optimal algorithm in hindsight, as they aim to use up the capacity around 2,440 refugees rather than the 2,310 who ended up arriving. This effect is reflected in the average employment rates (top right), which lie below that of the optimal

[10]This means that the comparison to the historical algorithm is not quite on equal terms, since the latter is constrained by a different set of capacities. In all fiscal years except for 2017 and 2018, the final capacities are element-wise larger than the original capacities.

[11]Note that, due to the triangle smoothing, the drop starts dragging down the curve 500 arrivals before its actual start.

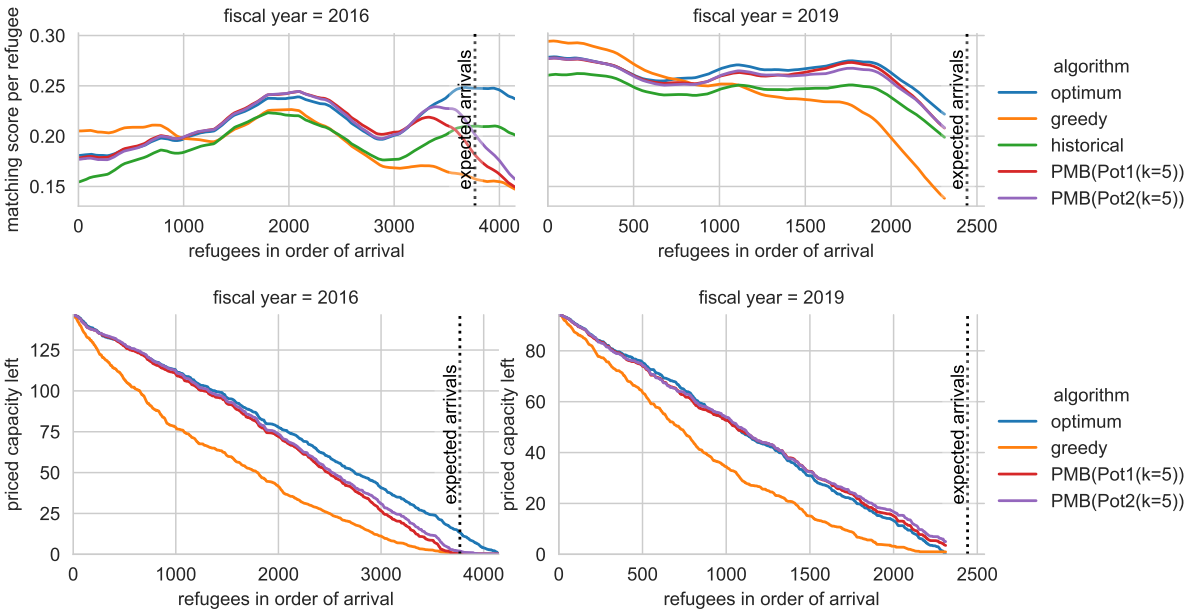


Figure 2.7: Evolution of the per-refugee match score and remaining priced capacity in order of arrival, for fiscal years 2016 and 2019 and one run per algorithm in the experiment of fig. 2.6 (whole cases, batches, initial capacities, potential algorithms do not know n). Dotted line show how many refugees the potential algorithms expect. Smoothing as in fig. 2.2. Priced capacity is not shown for historical since it uses different capacities.

algorithm throughout most of the year.[12]

The fiscal years of 2017 and 2018 stand out due to the fact that the total number of arriving refugees fell far short of the announced number reflected in the approved capacities: in 2017, arrivals amounted to 65% of the approved capacities, while they amounted to only 46% in 2018. Both of these years fall into the beginning of the Trump administration, which not only sharply reduced the announced intake of resettled refugees, but furthermore abruptly halted the intake of refugees from six predominantly Muslim countries starting from early 2017.

As the potential algorithm depicted in fig. 2.8 severely overestimates how many cases will arrive, it holds back much more priced capacity than would be optimal (bottom, solid lines). This causes the potential algorithms to extract less employment throughout the year than the optimal algorithm (top, solid lines). As observed in section 2.4.2, the capacities in 2018 are so loose that the greedy algorithm performs close to optimal.

In these two years, the US Department of State eventually reacted by correcting the expected arrivals downward and instructing the resettlement agencies to reduce their capacities. In fiscal year 2017, this revision came quite late and ended up underestimating the arrivals: where the arrivals amounted to only 65% of the initial capacities, they exceeded the revised total capacity at a level of 103%, rather than amounting to the 91% that was intended. Even if imperfect, this signal that arrivals are much lower than originally announced is still useful to the potential algorithms. Indeed, in fig. 2.8, the dashed curve corresponds to a potential algorithm that still starts out expecting 91% of the initial capacities to arrive, but expects only

[12]The drop in employment probabilities at the end of the fiscal year affects all algorithms including the hindsight optimum and must therefore be caused by an anomaly in arrival characteristics.

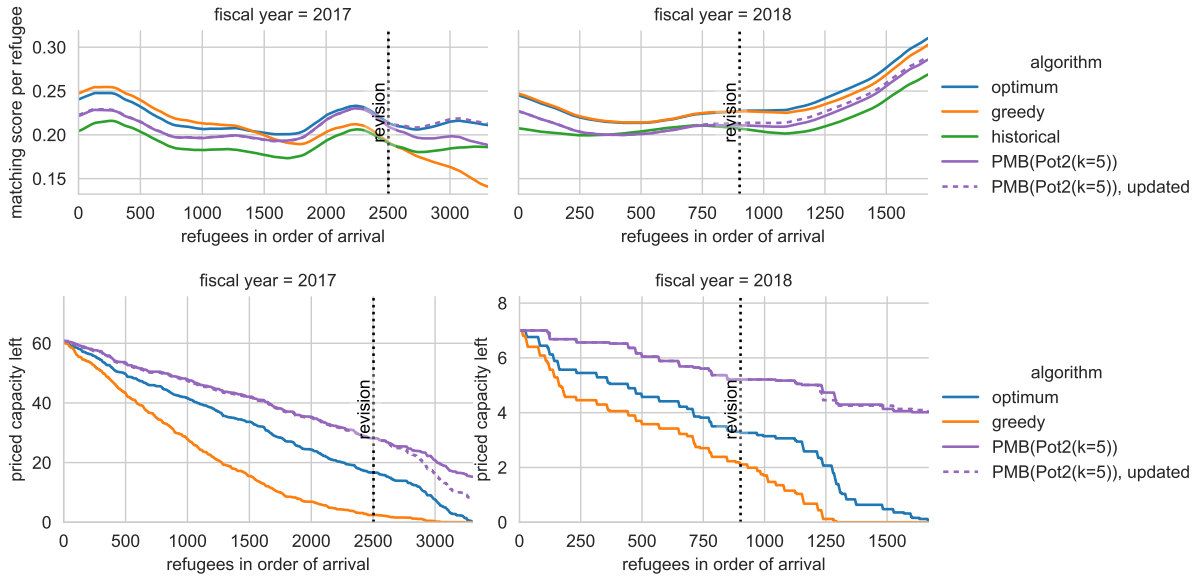


Figure 2.8: Evolution of the per-refugee match score and remaining priced capacity in order of arrival, for fiscal years 2017 and 2018 in the experiment of fig. 2.6 (whole cases, batches, initial capacities, potential algorithms do not know n). Dashed line shows evolution if potential algorithm updates its expected arrival number at time of capacity revision (dotted line).

91% of the revised capacities to arrive from the point on where they were announced (vertical line). While this information comes late, the algorithm in fiscal year 2017 uses the new information to burn through the remaining priced capacity more aggressively (bottom left), which allows for higher employment among refugees arriving after the revision of arrival numbers (top left). As a result, the employment reaches 97% of the optimum in hindsight, exceeding the value of 95% without the updated information that we showed in fig. 2.6.

By contrast, the revision in fiscal year 2018 did not yield much useful information; whereas the arrivals amounted to 46% of the initial capacities, they still amounted to 48% of the revised capacities. This seems to indicate that, even after half of the fiscal year's refugees had already been allocated, the administration overestimated the number of arriving refugees by a factor of two. Because the revision barely changed the number of expected arrivals, giving the potential algorithm access to this revised information does not have much effect (fig. 2.8, right).

While we have considered the informational value of revisions above, our experiments have not considered that these revisions actually reduced the allowable capacities. Although we include a variant of the experiment in appendix I.0.3, it is difficult to meaningfully compare the employment achieved by different algorithms if the parameters of the matching problem are changed so drastically during the matching period. One particular challenge is that, while the amount of reduction was extraneously decided, HIAS was involved in deciding which capacities to decrease, which was done in a way that depended on previous allocation decisions.[13] Since we only know the revised capacities that were agreed upon, not the counterfactual revision of capacities that would be made, the greedy algorithm and the potential algorithms might have already exceeded a reduced capacity before it was announced. This means that the

[13]While the sum of capacities did not change much in fiscal year 2018, the capacities of some affiliates were substantially decreased and those of others were substantially increased.

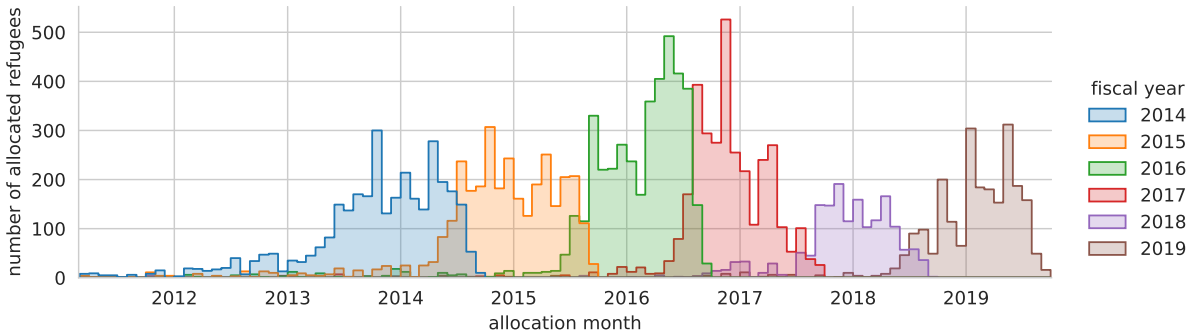


Figure 2.9: Monthly number of allocated refugees, disaggregated by fiscal year of arrival.

experiment rewards algorithms for greedily using up the capacity in the best affiliates before the revision, which we do not expect to be a good policy in practice. More generally, a substantial change in capacities is an exceptional situation, outside of our model, and cannot be addressed by our algorithm alone without manual intervention.

2.7.2 Better Knowledge of Future Arrivals

In the previous subsection, we demonstrated that, even without outside supervision, our potential algorithms lead to substantial improvements in total employment unless the announced capacities miss the eventual arrival numbers by a wide margin. Even in these typical years, however, more accurate arrival predictions could increase the total employment on the order of percentage points of the hindsight optimum. Obviously, more accurate information about arrivals would be even more useful in years like 2017 and 2018, in which the official information is unreliable.

One approach would be to use time-series prediction to estimate the number of arrivals. For instance, when the US Department of State revised the capacities for the fiscal year 2018 in January 2018 (several months into the fiscal year), the announcement that 2.5 times more refugees were still to come than had already arrived might have raised some doubts. However, the graph of monthly arrivals in fig. 2.9 shows that late increases in arrival rates may actually happen as they did in fiscal year 2016.[14]

A fundamental challenge that any data-driven approach faces is that there is very little data to learn from. Indeed, while HIAS has data on hundreds of thousand of refugees, they only have data on 15 fiscal years, which is, moreover, incomplete and smaller-scale in earlier years. Thus, there is a limited foundation to learn about how arrival patterns change between years. This task becomes especially difficult given that arrival numbers are heavily influenced by external events such as elections, the emergence of humanitarian disasters, and by changes in immigration policy, which cannot be deduced from past arrival patterns. Thus, while a time-series prediction approach might lead to marginal improvements over naïvely expecting 91% of the capacity to arrive, past arrival numbers are unlikely to give enough information to accurately predict future arrival numbers.

Fortunately, resettlement agencies such as HIAS already possess much richer information

[14]In fiscal year 2016, the number of arrivals after January 2016 was 1.6 times larger than the number that had arrived so far. In the fiscal year of 2015, the number of refugees arriving after January 2015 was only 75% of that arriving before.

and insights into the dynamics of refugee arrivals than a pure data approach would consider. In fiscal year 2017, for example, HIAS foresaw a worsening climate for refugee resettlement immediately after the November 2016 election[15] and was aware of concrete plans to drastically reduce refugee intake in January 2017,[16] both before these changes were reflected in arrival numbers and before the capacities were officially updated in March 2017. Similarly, HIAS continuously monitors domestic politics and international crises for their potential impact on resettlement, and moreover it has some limited insight into the resettlement pipeline, which allows it to prepare for changes in arrivals. We therefore believe that, rather than building a sophisticated tool for predicting arrivals in a fully autonomous manner, it is preferable to allow HIAS staff to override our prediction with more advanced information.

2.8 Implementation in *Annie*™ Moore

To enable HIAS to benefit from dynamic allocation via potentials, we have integrated new features into its matching software *Annie*™ Moore. A crucial design requirement is that HIAS staff must be able to override the allocation recommendations of *Annie*™ when they are aware of requirements outside of our model. From an interface-design perspective, the challenge is to visualize the effect of such overrides on total employment, enabling HIAS staff to make informed trade-offs. In the original, static model, this was easy enough: as the quality of a matching was just the total employment of the current batch, the interface labeled each case–locality match with its associated employment score, and staff could drag the case to other localities to see the respective employment scores. In a dynamic setting, however, presenting only the employment scores may unintentionally encourage HIAS staff to greedily use capacity in their overrides, at the expense of future arrivals.

As we illustrate in fig. 2.10, the new interface of *Annie*™ augments the original interface with information about affiliate potentials, thereby taking future arrivals into account. Specifically, the background color of the tile for case i encodes the *adjusted* employment score, that is, the original employment score $u_{i,\ell}$ less the value $s_i\, p_\ell$ of the total capacity consumed in affiliate ℓ.[17] The fact that the algorithm PMB always maximizes the sum of adjusted employment scores in its allocation of the current batch means that the algorithm is explainable in terms of the information presented to the user. In the interface, the *green* color spectrum indicates positive adjusted employment scores (meaning that the employment score of the case outweighs the loss in future employment), while the *red* color spectrum highlights negative adjusted scores (where a placement reduces future employment by more than its employment score). Darker colors signify greater magnitudes.

In overriding the allocation recommended by *Annie*™, HIAS staff should be able to quickly find alternative placements for a case that do not reduce immediate and future employment by more than necessary. To support this workflow, our interface shows the adjusted employment scores of a case across all affiliates at a glance: as shown in fig. 2.11, upon dragging a particular case tile from its current placement, all other case tiles temporarily fade in appearance, and the shading of every affiliate tile temporarily assumes the adjusted employment score relative to the selected case. By hovering a selected case tile over a new affiliate, the original (numeric)

[15] https://www.hias.org/news/press-releases/hias-calls-president-elect-trump-respect-longstanding-refugee-policy

[16] https://www.hias.org/news/press-releases/trumps-planned-action-refugees-betrayal-american-values

[17] The employment scores of cases in affiliates are prominently retained in a text label.

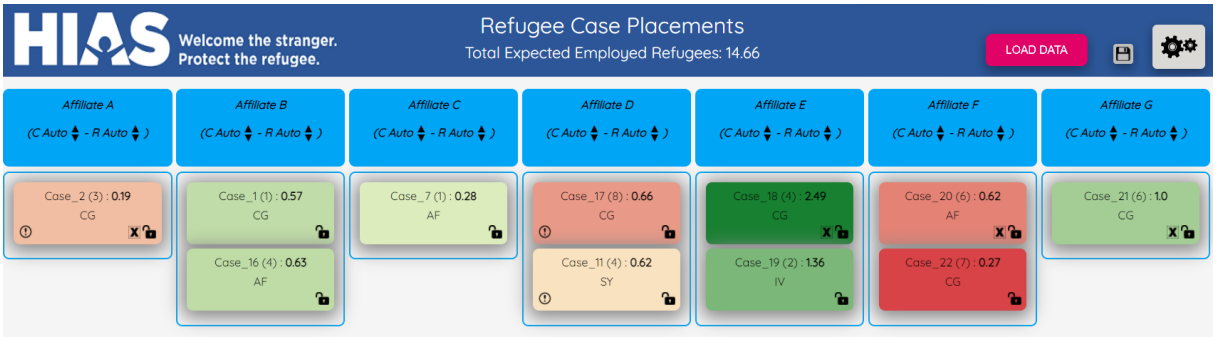


Figure 2.10: Updated *Annie*™ Interface. Family tiles now show both original numerical employment scores of families in affiliates, as well as the *adjusted* employment score by its shading. Green indicates positive adjusted scores, red negative scores, and darker colors represent greater magnitudes.

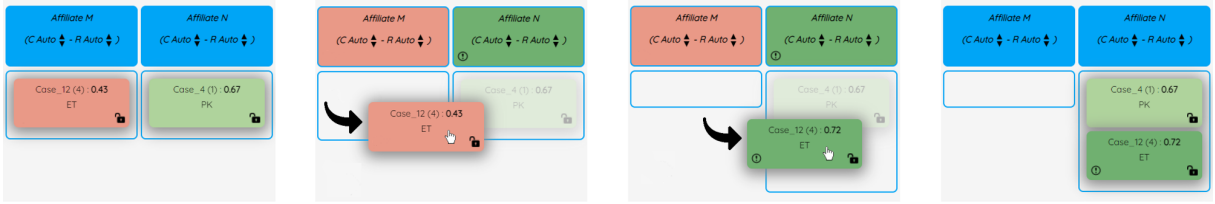


Figure 2.11: Moving a family tile. Other case tiles fade, and affiliate tiles are colored as per their adjusted employment scores in shades of green (positive) or red (negative). Exclamation marks indicate incompatibilities.

employment score and the adjusted match score (background color of the case tile) dynamically update. Moreover, incompatibilities with affiliates due to nationality, language, family size, and single parent households can be seen via an exclamation mark in the lower left corner of the affiliate tile. After dropping the case tile in a new affiliate, the background color for each affiliate returns to its original blue shade, and all affiliate-tile exclamation marks disappear.

On a separate screen (not shown), *Annie*™ enables the entry of a prediction for total refugee arrivals, as mentioned in section 2.7.2. This estimate can be critical to inform the process of estimating proper shadow prices, as at times HIAS is in a better position to give more accurate case arrival predictions than officially announced capacities.

2.9 Conclusion

We have developed and implemented algorithms for dynamically allocating refugees in a way that promotes refugees' prospects of finding employment. These algorithms are grounded in theory and perform very well when tested on real data.

While we have tested the algorithms as an autonomous system, the success of *Annie*™ in increasing employment outcomes in practice will depend on how it performs in interaction with HIAS resettlement staff. In section 2.7.2, we already saw that the allocation decisions of *Annie*™ can greatly profit from human users providing better estimates of future arrivals. Human input is equally crucial in dealing with uncertainty in several other places: for example, HIAS staff might intervene by correcting the arrival year of a case if the Department of State's

estimate seems off, or they might increase some affiliates' capacities late in the year if they anticipate that these capacities can be increased. By allowing all parameters of the matching problem to be changed, *Annie*™ allows HIAS resettlement staff to improve the matching using all available information.

Ideally, the human-in-the-loop system consisting of the matching algorithm and HIAS staff can combine the strengths of both of its parts: On the one hand, the algorithms in *Annie*™ capitalize on subtle patterns in employment data and manage capacity more effectively over the course of the fiscal year. On the other hand, the expert knowledge of HIAS staff enables the system to handle the uncertainty that is inherent in a matching problem involving the actions of multiple government agencies, dozens of affiliates, and thousands of refugees. In light of the current administration's plan to increase refugee intake from 15,000 to 125,000,[18] we foresee both parts playing a crucial role: the increasing scale of the problem will make data-based algorithms more effective, and human guidance will be necessary to navigate the evolving environment of a rapidly growing operation.

[18]https://www.hias.org/news/press-releases/biden-administration-announcements-refugees-are-welcome-news

Chapter 3

Risk-Averse Placement Optimization in Refugee Resettlement

Refugees are resettled into communities in many ways, and more recently with the carefully designed analytical approaches. The refugee placement optimization software *Annie*™ Moore estimates refugee-locality match quality scores using predictive modeling of past refugee placement and outcomes data to generate the likelihood of employment for incoming refugees. While estimated scores are used for offline optimal matching of arriving refugees in subsequent placement periods, inherent uncertainty exists with respect to the quality score estimation. This uncertainty can lead to different optimized outcomes that risks adverse effects on refugee welfare. We explicitly incorporate risk into the optimization of refugee outcomes and propose new methods to hedge against this risk, while retaining a majority of the total expected employment.

3.1 Introduction

This research is an extension to our earlier developments in using analytics to improve refugee resettlement [61], where we addressed the operational challenge of matching refugee families (cases) to local communities (affiliates) using machine learning and integer optimization.

At the end of 2020, there were 20.7 million refugees under the mandate of the United Nations High Commission for Refugees (UNHCR). Refugees are forcibly displaced people who have crossed international borders and mostly live within cities or established camps in the countries of asylum which are mostly countries neighbouring their country of origin. Of those, an estimated of 1.44 million refugees were projected to be in need of *resettlement*–permanent relocation from their asylum country to a third country. These are the most at risk refugees due to medical conditions or other urgent protection needs. The United States has historically been the world's largest destination of resettled refugees. Before arrival into the United States, refugees are assigned to one of nine national resettlement agencies which are responsible for placing refugee families into their own network of affiliates. Each affiliate has limited capacity on different provided resources and each family has specific needs that should be served by the affiliate. Historically, the decision-making on best initial family-affiliate match was done

largely manually with multiple sources of inefficiencies. We refer the readers to [61] for the trough discussion regarding inefficiencies in the manual matching process and advantages of using analytics to augment decision making in refugee resettlement.

Throughout this chapter we use the term ***outcome*** repeatedly. While traditionally *outcome* represents an optimal solution to an optimization model, in this chapter *outcome refers to* the *likelihood of employment* associated with an optimal match between refugee family and affiliate, also known as *family-affiliate match quality score*. We also use the term *optimal outcome(s)* and *optimization outcome(s)* to refer to either families' employment likelihood at optimal placements or the total (optimal) employment obtained by maximizing the objective function value.

[61] first use machine learning to estimate the likelihood of employment of refugee families in the affiliates. It introduces an optimization model that takes a group of refugee families and a set of affiliates as well as estimated employment outcome of refugee families in affiliates and optimally places refugee families in affiliates, with the objective of maximizing the total employment while respecting existing constraints including affiliates capacity constraints. [75] extends the original optimization model to a dynamic allocation system based on two-stage stochastic programming to improve employment outcomes of refugee families.

Both optimization models interpreted the estimated employment outcomes used as match quality scores in the objective function coefficients are known *accurately*. These models are most meaningful when the decision maker is risk-neutral. There exists inherent *error* with respect to the estimation of employment probabilities that causes uncertainty with respect to optimization outcomes. This uncertainty or variation of optimization outcomes is interpreted as risk. To provide a proper solution for risk-averse decision makers, an optimization model should account for the potential impact associated with risk based on risk preferences of decision makers. The omission of risk from an optimization model may result in optimal placement decisions obtained with respect to expected outcomes that may unfortunately permit disruptive variations on these outcomes when different estimations are realized [76].

Risk in optimization problems have been studied from a variety of perspectives. However, the related literature mainly interprets risk as expressed by uncertainty around total expected objective function value (***Collective*** risk). Considering that the optimal employment outcome of each refugee family is no less important than the total optimal employment, we introduce an alternative notion of risk at the individual refugee family level (***Individual*** risk) that properly accounts for vulnerability of refugee families and fits well in the context of refugee resettlement. Our definition of risk is concerned with uncertainty around the employment outcome of each refugee family at optimal placement. We seek to mitigate this risk from an optimization point of view.

We first formulate refugee resettlement optimization model in classical collective-level risk-averse setting. We then propose a new mathematical model that considers the risk at the individual level. Then we compare the outcomes of these two models and demonstrate how our proposed model appropriately optimizes the placements while hedging against the family-level risk. We then address fairness in refugee resettlement by proposing a scheme that assigns relatively less risk to those refugee families that are the most vulnerable. We develop unsupervised learning methods, informed by supervised learning outcomes (namely employment likelihoods), to cluster families, and use these clusters in our optimization modeling to redistribute relatively greater family-level risk to the those families exhibiting the least vulnerability.

3.2 Background and Related Work

Optimization under uncertainty occurs when one or more parameters involved in the model are subject to randomness. This may be due to variety of reasons, including estimation error and unexpected disturbances [77]. These types of problems have attracted tremendous attention in a wide variety of research contexts. Several class of optimization frameworks have been developed and analyzed to address uncertainty, under the broader name of *stochastic optimization*, including two- or multi-stage stochastic programming [78–81], Chance-constrained programming [82–84], robust optimization [85–88]. These different modeling approaches can be used alone or in combination with one another, reflecting different aspects of the applied problem at hand.

The shortcomings of optimizing the uncertain outcomes *on average* while neglecting consideration of the fluctuation of specific outcome realizations, motivated the development of risk averse optimization [89–91]. Modeling risk aversion in optimization requires risk to be *measured*. In these models, an uncertain outcome is characterized by the expected outcome or mean together with the risk that measures the uncertainty of the outcome. Driving the key trade-off between risk and mean, this class of models optimize for the expected outcome while protecting against possible losses. A popular risk-averse approach is the mean–risk model [92], where for a given value of the mean we minimize the risk or for a given value of risk we optimize the mean. The level of trade-off between mean and risk can be controlled by a parameter that reflects the degree of the decision maker's risk aversion. Many reliable risk measures have been developed and are widely used in academia and industry including variance, absolute error, deviation or semi-deviation from mean or target value, value-at-risk and conditional value-at-risk. These risk measures may be included in the objective or expressed as constraints.

3.3 Placement Optimization in Refugee Resettlement

[61] formulate the operational challenge of optimally placing refugee families to affiliates using integer optimization. The goal of the optimization model is to place families into affiliates in a way that it maximizes total employment while respecting the affiliates' capacities and families' needs. We briefly explain the model here. Table 3.1 presents the notation.

Table 3.1: Notation for Refugee Resettlement Placement Optimization

Parameter	Definition
$\mathcal{F}$	Set of refugee families, indexed by i
$\mathcal{L}$	Set of local communities or affiliates, indexed by ℓ
s^i	Size of family $i \in \mathcal{F}$
c_ℓ	Capacity of affiliate $\ell \in \mathcal{L}$
a_ℓ^i	Binary indicator for family $i \in \mathcal{F}$ and affiliate $\ell \in \mathcal{L}$; 1 if compatible, 0 otherwise
x_ℓ^i	Decision variable equals 1 if family i is matched to affiliate ℓ, and 0 otherwise
v_ℓ^i	Match quality score of family $i \in \mathcal{F}$ to affiliate $\ell \in \mathcal{L}$

Baseline Placement Optimization Model.

$$\text{maximize} \quad \sum_{i=1}^{|\mathcal{F}|} \sum_{\ell=1}^{|\mathcal{L}|} v_\ell^i x_\ell^i \tag{3.1a}$$

$$\text{s.t.} \quad \sum_{\ell=1}^{|\mathcal{L}|} x_\ell^i = 1, \ \forall\, i, \tag{3.1b}$$

$$\sum_{i=1}^{|\mathcal{F}|} s^i x_\ell^i \leq c_\ell, \ \forall\, \ell, \tag{3.1c}$$

$$x_\ell^i \leq a_\ell^i, \ \forall\, i, \ \forall\, \ell, \tag{3.1d}$$

$$x_\ell^i \in \{0, 1\}, \ \forall\, i, \ \forall\, \ell. \tag{3.1e}$$

Objective function (3.1a) maximizes the total of employment probabilities of all families subject to *allocation constraints* (3.1b)–(3.1e). Constraint set (3.1b) ensures that each family is assigned to one affiliate. To ensure that the optimization problem is always feasible, we assume that the model contains a special *unplaced* affiliate with unbounded capacity corresponding to leaving families unmatched; any family matched to the unplaced affiliate contributes zero to the objective function (no likelihood of employment), and in reality is subsequently addressed through manual measures. Constraint set (3.1c) ensures that upper bounds on capacities are respected for all affiliates. Constraint set (3.1d) ensures that family-affiliate matches can only occur when the affiliate can support the needs of the family. Variable domains are specified in (3.1e).

3.3.1 Uncertainty in Employment Estimates

Formulation (3.1) depends upon the uncertain and estimated likelihoods of employment for every refugee family and affiliate. However, for each family-affiliate match score, there is a probability distribution of predicted values around the most likely point estimate. We use bootstrapping to generate the distribution of predictions. Table 3.2 provides the notation used in this and future sections. Between fiscal year 2010 and 2016, we have around 5000 refugees arrived at HIAS[1], of which 2,486 refugees were in the working age range (between 18 and 65 years old). These working-age refugees form our training dataset. For K iterations, we re-sample with replacement the training dataset and on each new sampled dataset, we re-estimate a new predictive model. Each predictive model estimates a new set of employment probabilities for refugee families in the test dataset in eight main affiliates. Our test dataset includes refugees arrived in fiscal year 2017 (839 refugees / 329 families). We refer to each set of predictions as a *bootstrap instance* or *scenario*. Let $\mathcal{K}$ be the set of scenarios generated from bootstrap procedure ($|\mathcal{K}| = K$). Each scenario $k \in \mathcal{K}$ contains alternative sets of family-affiliate match quality score estimations. Let $\mathcal{F}$ denote a set of refugee families and $\mathcal{L}$ denote a set of affiliates. In this case, we let v_ℓ^{ik} be the match quality score of family $i \in \mathcal{F}$ to affiliate $\ell \in \mathcal{L}$ in scenario $k \in \mathcal{K}$. To optimize refugee placements in model (3.1), we use the mean (expected) match quality scores $\bar{v}_\ell^i = {}^1\!/\!_{|\mathcal{K}|} \sum_{k \in \mathcal{K}} v_\ell^{ik}$ in the objective function. This is known as

[1]US resettlement agency HIAS (founded as the Hebrew Immigrant Aid Society). Please refer to Chapter 1.2.1 for more information about HIAS and refugee resettlement in the United States

the total expected value problem [81]. As depicted in Figure 3.1, for each family-affiliate pair the match quality scores v_ℓ^{ik} are averaged over all scenarios to obtain $\bar{v}_\ell^i$.

Table 3.2: Additional Notation for Risk-Averse Refugee Resettlement Optimization Model

Parameter	Definition
$\mathcal{K}$	Set of scenarios, indexed by k
v_ℓ^{ik}	Match quality score of family $i \in \mathcal{F}$ to affiliate $\ell \in \mathcal{L}$ in scenario $k \in \mathcal{K}$
$\bar{v}_\ell^i$	Mean match quality score of family $i \in \mathcal{F}$ to affiliate $\ell \in \mathcal{L}$ over all $\lvert\mathcal{K}\rvert$ scenarios

Figure 3.1: Each panel represents a bootstrap instance or scenario, contains a set of family-affiliate match quality scores estimation. For each family-affiliate pair we average match quality scores over all scenarios to get $\bar{v}_\ell^i$.

3.3.2 Risk in Placement Optimization

Based on which scenario is realized, optimization outcomes will likely differ from the total expected value problem outcomes. This variation in the optimized outcomes around expected values implies risk and can be addressed in at least two directions. First, there is variation around the total expected objective function value which means uncertainty about the total expected employment from maximizing all family-affiliate placements. We refer to this type of risk as *collective-level* risk. Second, there is variation around expected employment outcome of each refugee family which means uncertainty about every family's employment outcome at its optimal placement. We refer to this type of risk as *individual-level* risk. While the collective-level risk is well studied in the literature and many optimization models have been proposed to address this risk including well-known mean-risk models, these models fail to address the individual-level risk that is concerned with uncertainty around the employment outcome at the optimal location for each family. In what follows, we first formulate the refugee resettlement model according to collective-level risk that is we embed the refugee resettlement

optimization model into the well-studied mean-risk models in the literature with risk related to total employment. Then we propose our modeling approach to hedge against the individual-level risk that is the risk-hedging model with risk defined in family level. Finally we compare and contrast the performance of these two models with respect to family level risk reduction.

3.4 Risk-Averse Placement Optimization

In this section we study risk in placement optimization. We begin with a classical perspective that considers risk in the *collective*, and then introduce a perspective that emphasizes the *individual* risk. We show how this new perspective better mitigates the risk faced by the most vulnerable of refugees, and proceed to build upon this foundation to develop classification strategies for ensuring fairness across categories of refugee employability.

3.4.1 Collective Risk Aversion

Mean-risk models are well-studied in the literature and allow for the hedging of risk that is connected to the variation of the total objective function value. Markowitz [92] first analyzed the mean-risk model by using variance as the measure of risk. In the mean-variance model, risk is incorporated in the objective function through penalization of variance from the mean. Following the seminal work of Markowitz, we incorporate this mean-risk notion into our refugee resettlement optimization model in formulation (3.2).

Collective Risk-Averse Refugee Resettlement Model (Collective-RARR):

$$\text{maximize} \quad \bar{q} - \rho \frac{1}{|\mathcal{K}|} \sum_{k=1}^{|\mathcal{K}|} \left(\bar{q} - q^k\right)^2 \tag{3.2a}$$

$$\text{s.t.} \quad \bar{q} = \sum_{i=1}^{|\mathcal{F}|} \sum_{\ell=1}^{|\mathcal{L}|} \bar{v}_\ell^i x_\ell^i, \tag{3.2b}$$

$$q^k = \sum_{i=1}^{|\mathcal{F}|} \sum_{\ell=1}^{|\mathcal{L}|} v_\ell^{ik} x_\ell^i, \ \forall\, k, \tag{3.2c}$$

Allocation Constraints (3.1b)–(3.1e).

This model maximizes the trade-off between total expected employment $\bar{q}$ and risk interpreted as the variance around the total expected value and is expressed as the penalized term in objective (3.2a). The level of trade-off is controlled by parameter ρ. When ρ equals zero we obtain the highest possible value for $\bar{q}$ at optimal placements. By increasing ρ and thereby placing greater emphasis on risk, the optimal placements may change to decrease the variance. While in this way a lower optimal value is obtained for $\bar{q}$, the variance around this value is also reduced.

3.4.2 Individual Risk Aversion

While formulation (3.2) addresses risk in the collective through rewarding placements that hew closer to the mean, it does so without particular regard to individual refugee families. This is

problematic when considering that some refugees are particularly prone to poor employment outcomes. We thus propose a novel risk measure for refugee resettlement, as represented in formulation (3.3). In this model the risk is defined as the cumulative variance around optimal outcome for every refugee family, w^i. The risk in this model capture the family-level employment variation and is included in the objective function as a penalty. The penalty term is weighted by a control parameter ρ that modulates the importance of the total expected employment outcomes versus family-level risks.

Individual Risk-Averse Refugee Resettlement Model (Individual-RARR):

$$\text{maximize} \quad \bar{q} - \rho \sum_{i=1}^{|\mathcal{F}|} w^i \tag{3.3a}$$

$$\text{s.t.} \quad \bar{q} = \sum_{i=1}^{|\mathcal{F}|} \sum_{\ell=1}^{|\mathcal{L}|} \bar{v}_\ell^i x_\ell^i, \tag{3.3b}$$

$$w^i = \frac{1}{|\mathcal{K}|} \sum_{k=1}^{|\mathcal{K}|} \left(\sum_{\ell=1}^{|\mathcal{L}|} \left(\bar{v}_\ell^i - v_\ell^{ik} \right) x_\ell^i \right)^2, \ \forall\, i, \tag{3.3c}$$

Allocation Constraints (3.1b)–(3.1e).

While the objective is to place refugee families in locations where their $\bar{v}_\ell^i$ is higher, to discourage larger values of w^i, the optimization model is forced to put more emphasis on optimal placement of families to reduce the risk for every family. In the next section we explain the effectiveness of the proposed improvements in Individual-RARR regarding family-level risk reduction and compare it to the Collective-RARR.

3.4.3 Collective Risk, versus Individual Risk

In this section we evaluate the performance of above models. We test the effectiveness of the proposed improvements in Individual-RARR regarding family-level risk reduction and compare it to the Collective-RARR. We first explain the initial setup to test these models and then we discuss on our findings.

Experimental Setup: We solve the Individual-RARR and Collective-RARR on real refugee data including 329 refugee families (839 refugees) arrived in 2017, eight locations and 100 scenarios. We step ρ values for both Individual-RARR and Collective-RARR models, across a range of small non-negative values and report the change in total optimal expected employment $\bar{q}$ as well as the change in family-level risk. We introduce a family-level risk metric that captures the mean of the variance of refugee employment outcomes at their optimal placements obtained by Individual-RARR and Collective-RARR. Let $\hat{\ell}(i)$ denote the optimal location for refugee i and $v_{\hat{\ell}(i)}^{ik}$ be the match score of refugee i in scenario k in optimal location $\hat{\ell}(i)$. Similarly let $\bar{v}_{\hat{\ell}(i)}^{i}$ be the expected match score of refugee i in optimal location $\hat{\ell}(i)$.

Definition 1 *Family-Level Risk Metric: The measurement of the variance of the employment*

outcome of family i at its optimal placement is defined by:

$$\frac{1}{|\mathcal{K}|}\sum_{k=1}^{|\mathcal{K}|}\left(\bar{v}^{i}_{\hat{\ell}(i)}-v^{ik}_{\hat{\ell}(i)}\right)^2 \tag{3.4}$$

Definition 2 *Mean Family-Level Risk Metric: The measurement of the mean of the families employment outcome variances at their optimal placements is defined by:*

$$\frac{1}{|\mathcal{F}|}\sum_{i=1}^{|\mathcal{F}|}\left(\frac{1}{|\mathcal{K}|}\sum_{k=1}^{|\mathcal{K}|}\left(\bar{v}^{i}_{\hat{\ell}(i)}-v^{ik}_{\hat{\ell}(i)}\right)^2\right) \tag{3.5}$$

Experimental Results: Figure 3.2 depicts the employment-risk trade-off for Individual-RARR and Collective-RARR for different values of ρ for each model. The vertical axis shows the total optimal expected employment $\bar{q}$ in (3.2a) and (3.3a) and the horizontal axis shows the family-level risk (3.5) for two models, all based on percentage of maximum employment and risk that is achievable at risk-neutral setting where $\rho = 0$ (top-right point in the graph). When we increase ρ for each model, both the total employment and risk decrease. For each level of loss on the total employment, the risk reduction is significantly higher on Individual-RARR compared to Collective-RARR which demonstrates the better performance of this model in hedging against the risk in the family-level. We discuss more on the performance comparison of Individual-RARR and Collective-RARR in Appendix J and Appendix K.

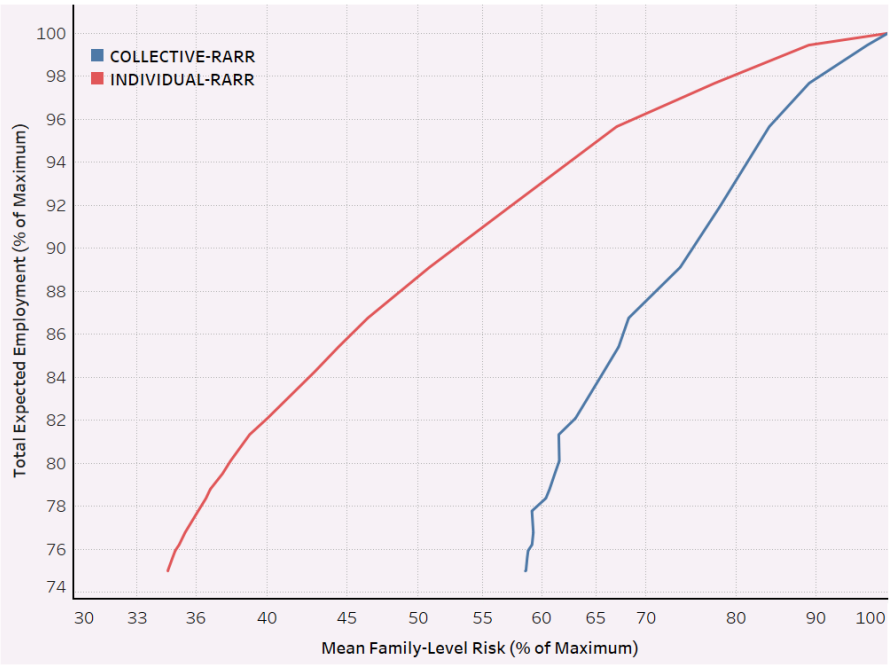


Figure 3.2: Employment-risk trade-off for Individual-RARR and Collective-RARR for different values of ρ for each model.

3.4.4 Fairness in Individual Risk Aversion

While INDIVIDUAL-RARR promises impressive results in terms of hedging against the family-level risk metrics, we now discuss an issue of fairness that is disregarded with this formulation. The horizontal axis in Figure 3.3 shows the refugee families and the vertical axis shows the risk of families formulated in (3.4). The blue line shows the risk at the optimal placements under INDIVIDUAL-RARR at risk-neutral setting ($\rho = 0$). The red line shows the risk at optimal placements under INDIVIDUAL-RARR at $\rho = 45$ and loss on total expected employment is about %21 of the total optimal employment at risk-neutral setting.
Lines are sorted based on the values on the blue line. The data points on the red line that are below the blue line are families that INDIVIDUAL-RARR at risk-averse setting placed them in less risky locations compared to risk-neutral setting. The points on the red line that are above the blue line are families that INDIVIDUAL-RARR at risk-averse setting placed them in more risky locations compared to risk-neutral setting. While risk is reduced for most of the families, it doesn't change or even is increased for others. Also the magnitude of risk reduction for families is different. This introduces an issue of *fairness* in risk reduction for refugee families.

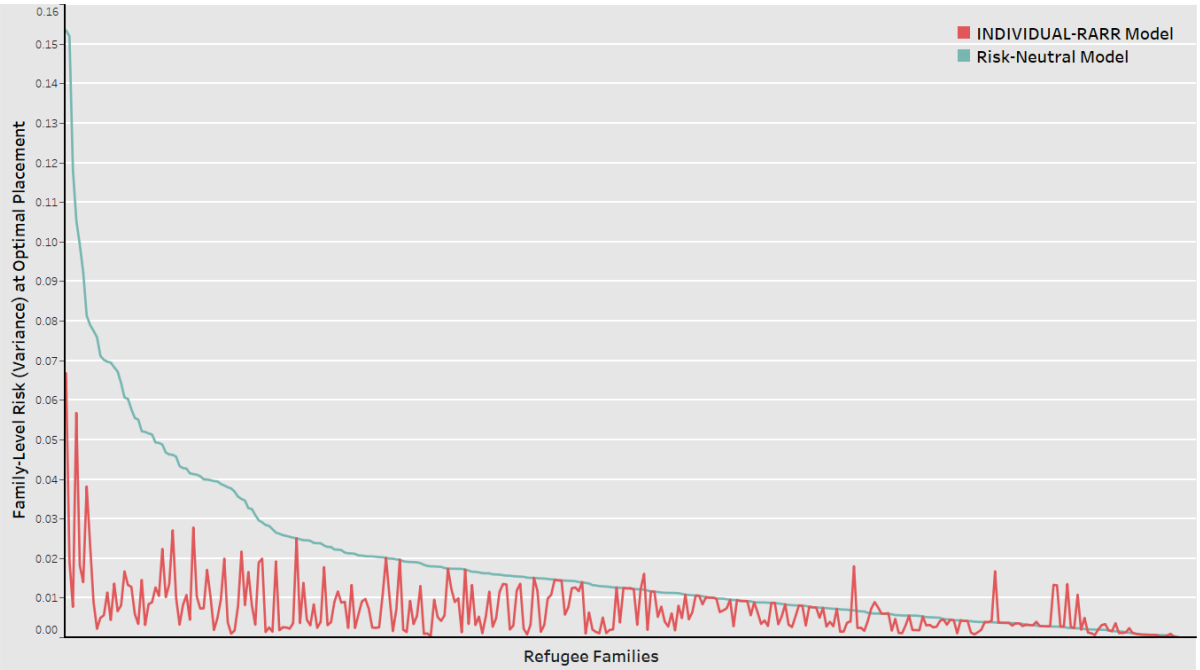


Figure 3.3: Risk of the refugee families formulated in (3.4), by solving INDIVIDUAL-RARR. Blue: risk-neutral setting ($\rho = 0$). Red: risk-averse setting ($\rho = 45$ and loss on total expected employment is about %21 of the total optimal employment of risk-neutral setting)

The uneven mean-risk trade-off in INDIVIDUAL-RARR model is also visualised in Figure 3.4 for six sample families across different values of ρ. Each panel depicts the outcome for each refugee family. The horizontal axes show $\rho = 0, 0.2$ and 0.5. Box and whisker plots depict the distribution of employment likelihood at optimal locations $v^{ik}_{\hat{\ell}(i)}$, across all scenarios. Red lines indicate $\bar{v}^{i}_{\hat{\ell}(i)}$, the mean employment at optimal location. It can be observed that the amount of trade-off can vary across families. This introduces another aspect of the issue of *fairness*. The reason for this uneven trade-off is that in objective function (3.3a), the sum of variances is penalized. Thus the variance at optimal placement does not evenly decrease over

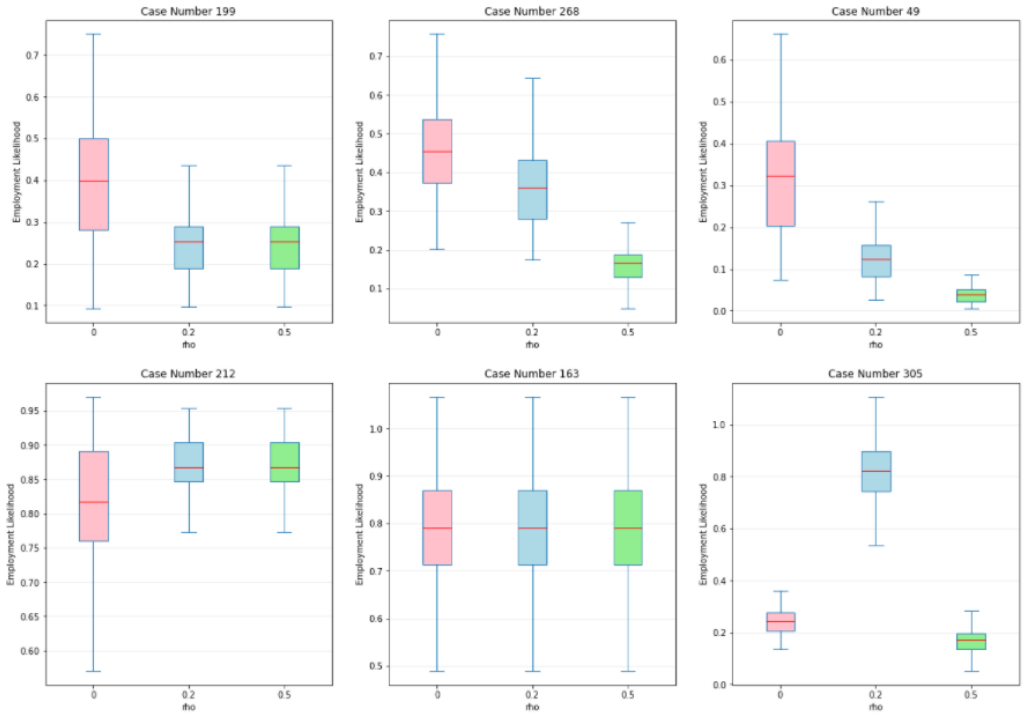


Figure 3.4: Mean-risk trade-off performance for six sample families across different values of ρ. Box and whisker plots depict the distribution of employment likelihood at optimal locations $v^{ik}_{\hat{\ell}(i)}$, across all scenarios. Red lines indicate $\bar{v}^{i}_{\hat{\ell}(i)}$, the mean employment at optimal location.

all families, unnecessarily favoring some families over others. The issue of fairness in risk aversion for families suggests that alternative formulations are needed to give us more control on addressing the level of risk for different refugee families.

3.5 Fair Risk-Averse Placement Optimization

To effect greater control on risk at optimal placements across families and consequently employ fair risk aversion, we propose an alternative, optimization-based approach that incorporates a risk-hedging mechanism into the constraint space. This mechanism directly limits family-level risk at optimal placements according to prespecified risk tolerance levels. Due to the large number of refugee families, however, choosing risk tolerance levels appropriate for every family is impractical.

Standard clustering approaches such as k-means clustering may be used to reduce the number of unique risk tolerance levels by grouping refugee families by similar features, such as the mean employment likelihoods of each family across all affiliates. All families within each cluster will then be represented by a single risk tolerance level, thereby limiting the risk for all families within each cluster. Let $\mathcal{C}_j, j \in \mathcal{J} = \{0, 1, 2, ...\}$, be a set of families that are clustered together. Any number of clusters may be chosen, where:

$$\mathcal{F} = \bigcup_{j \in \mathcal{J}} \mathcal{C}_j, \ \mathcal{J} = \{0, 1, 2, ...\}.$$

We form five clusters of families: low ($j = 0$), low-medium ($j = 1$), medium ($j = 2$), medium-

high ($j = 3$) and high ($j = 4$) employable families. The low employable cluster groups families that have relatively lower estimated employment probability in most locations and the high employable cluster groups families that have high estimated employment probability in most locations. Figure 3.5 depicts the clusters for two pairs of randomly chosen affiliates.

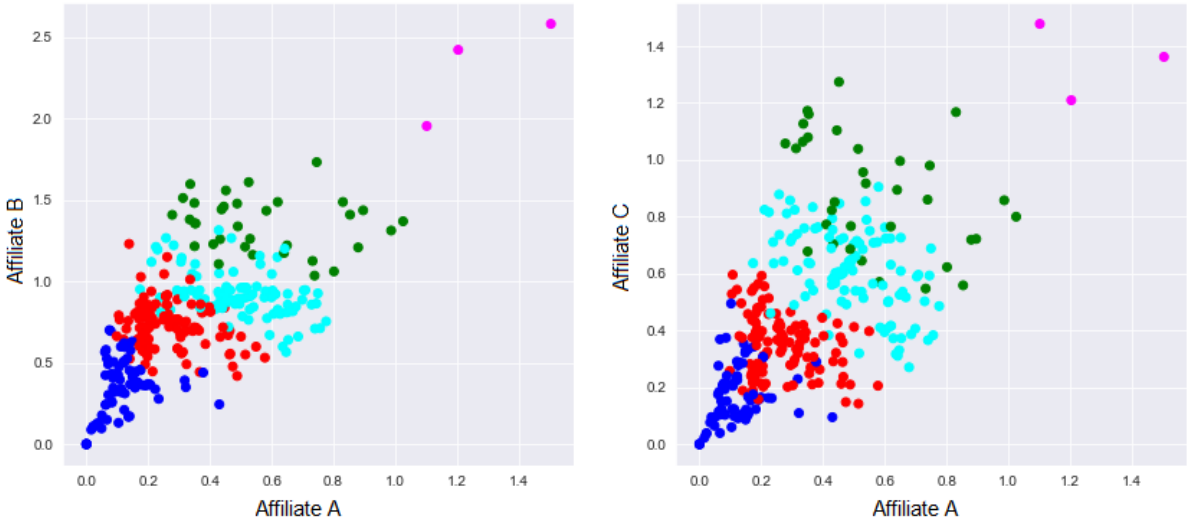


Figure 3.5: Cluster of refugee families based on their employability in all locations. The axes show the mean employment likelihood of families in sample locations. Blue: low employable ($j = 0$), Red: low-medium employable ($j = 1$), Cyan: medium employable ($j = 2$), Green: medium-high employable ($j = 3$) and Purple: high employable ($j = 4$)

Employing as a risk measure the symmetric variation around the mean as in (3.2a) and (3.3c) gives rise to other issues of fairness. Refugee families that have particularly low employment likelihoods across many locations may benefit from having relatively greater emphasis placed on limiting deviation below the mean at optimal placement, without hindering higher deviations above the mean. On the other hand, for refugee families that have higher employability, the deviation might be limited from both sides, with relatively more tolerance below, and less tolerance above. Thus, the ability to control risk on one or both sides of the mean is desirable in the context of refugee resettlement, suggesting the use of (semi-)deviation from the mean for risk management.

Formulation (3.6) accommodates the aforementioned concerns of symmetric variation around the mean by employing (semi-)deviation from the mean in the form of constraints, as applied to all families sharing the same employability cluster. That is, each cluster $\mathcal{C}_j$ has its deviation from the mean at the optimal placement upper (lower) bounded by parameter $\delta_R^{\mathcal{C}_j} \in \mathbb{R}_{\geq 0}$ ($\delta_L^{\mathcal{C}_j} \in \mathbb{R}_{\geq 0}$). This enables the limiting of deviation from the mean for every cluster $j \in \mathcal{J}$, every family $i \in \mathcal{C}_j$, and every scenario $k \in \mathcal{K}$.

Fair Risk-Averse Refugee Resettlement Model (Fair-RARR)

$$\text{maximize} \quad \sum_{i=1}^{|\mathcal{F}|} \sum_{\ell=1}^{|\mathcal{L}|} \bar{v}_\ell^i x_\ell^i \tag{3.6a}$$

$$\text{s.t.} \quad \sum_{\ell=1}^{|\mathcal{L}|} \left(v_\ell^{ik} \; - \; \bar{v}_\ell^i\right) x_\ell^i \leq \delta_R^{\mathcal{C}_j} \; , \; \forall \, j \in \mathcal{J}, \; \forall \, i \in \mathcal{C}_j \; , \; \forall \, k \in \mathcal{K}, \tag{3.6b}$$

$$\sum_{\ell=1}^{|\mathcal{L}|} \left(\bar{v}_\ell^i \; - \; v_\ell^{ik}\right) x_\ell^i \leq \delta_L^{\mathcal{C}_j} \; , \; \forall \, j \in \mathcal{J}, \; \forall \, i \in \mathcal{C}_j \; , \; \forall \, k \in \mathcal{K}, \tag{3.6c}$$

Allocation Constraints (3.1b)–(3.1e).

Formulation (3.6) empowers the hedging of risk across all families of each cluster $j \in \mathcal{J}$ through two parameters, $\delta_L^{\mathcal{C}_j}$ and $\delta_R^{\mathcal{C}_j}$. For example, for families in the low employment likelihood cluster, managing deviation below the mean may be critical to reducing unnecessary risk exposure and associated vulnerability; on the other hand, deviation above the mean presents little to no issue. This suggests a value of $\delta_R^{C_0} = \infty$, and a value of $\delta_L^{C_0}$ having a relatively small distance below the mean to induce a fairly conservative tolerance.

3.5.1 Parameter Selection for Fair Risk-Averse Model

There are myriad possible combinations for parameters $\delta_L^{\mathcal{C}_j}$ and $\delta_R^{\mathcal{C}_j}$ over all clusters $\mathcal{C}_j, j \in \mathcal{J} = \{0, 1, 2, 3, 4\}$; and, as they represent nonnegative continuous values used in solving the Fair-RARR model, it is necessary to judiciously examine the space of possible parameters. Given the interest in understanding the sensitivity of the model to changes in $\delta_L^{\mathcal{C}_j}$ and $\delta_R^{\mathcal{C}_j}$, we pursue a grid-based parameter space search by step-wise increasing $\delta_L^{\mathcal{C}_j}$ and $\delta_R^{\mathcal{C}_j}$, $\forall j \in \mathcal{J} = \{0, 1, 2, 3, 4\}$. While some small parameter values induces infeasibility in the Fair-RARR model, some ranges of larger values of $\delta_L^{\mathcal{C}_j}$ and $\delta_R^{\mathcal{C}_j}$ result in identical optimal solutions. We seek to identify the tolerance of these parameter values to define the boundaries of the parameter grid.

As an illustrative example, considering finding the lowest possible value for $\delta_L^{\mathcal{C}_j}$ for cluster $\mathcal{C}_j$ with a single family and a single location, relaxing for the sake of illustration all other constraint sets. There is a distribution of employment estimates over K scenarios for this family in this location. The lowest possible value for $\delta_L^{\mathcal{C}_j}$ that does not induce infeasibility is the difference between the mean employment estimate and the minimum employment estimate over all scenarios (that is, the worst-case scenario). Accordingly, we refer to this difference as the *worst-case left deviation* from the mean. More generally, consider assigning this family to one of multiple possible locations. For each location there is a distribution of estimations, each with a corresponding *worst-case left deviation*, and the value of $\delta_L^{\mathcal{C}_j}$ can be decreased to the minimum of these *worst-case left deviation* values. There is no need to set $\delta_L^{\mathcal{C}_j}$ beyond this minimum value, as then no locations could satisfy constraints (3.6c), thus causing infeasibility.

In reality, each cluster contains a group of families under consideration for many locations, under risk constraints (3.6b) and (3.6c) and allocation constraints (3.1b)–(3.1e). As each family has a *minimum worst-case left deviation*, then conservatively speaking, the lowest possible value of $\delta_L^{\mathcal{C}_j}$ for this cluster becomes the maximum of these minimum values. Hence, the maximum minimum worst-case left deviation across all families in that cluster determines

how much the value of $\delta_L^{\mathcal{C}_j}$ can be reduced. The complementary analysis is conducted to find the lowest possible value of $\delta_R^{\mathcal{C}_j}$ for this cluster, that is, the *maximum minimum worst-case right deviation* from the mean. This exercise can be repeated for all clusters individually to determine the lowest possible value for $\delta_L^{\mathcal{C}_j}$ and $\delta_R^{\mathcal{C}_j}$ over all clusters.

The FAIR-RARR model can be solved for every point in the parameter grid, each specifying unique risk constraints (3.6b) and (3.6c), followed by a post-optimality descriptive analysis to search for desired outcomes. This grid-based optimization procedure may result in exceedingly many solutions, thereby complicating the post-optimality analysis. We now describe some initiatives to reduce the feasible solution space by adding constraints to eliminate potentially undesirable solutions. For simplicity we set $\delta_R^{\mathcal{C}_j}$ to a sufficiently large value for all clusters $\mathcal{C}_j$, searching only to identify desirable values for $\delta_L^{\mathcal{C}_j}$.

3.5.2 Reducing Risk in Outcomes for Low Employable Refugees via Eliminating Inequity

The FAIR-RARR model can be augmented with additional constraints to further shape the feasible region toward inducing desirable outcomes. In this section we propose several possible constraint sets for the FAIR-RARR model to improve outcomes for vulnerable families.

While decreasing $\delta_L^{\mathcal{C}_j}$ and $\delta_R^{\mathcal{C}_j}$ and optimizing the FAIR-RARR model results in placing families in locations in which the risk around their employment is lower, this risk reduction is gained at the cost of loss in the expected employment. Clusters compete on this risk gain, employment loss trade-off. We are particularly interested in improved outcomes for vulnerable families in the low employable clusters, with respect to both risk gain and employment loss. To this end we introduce several new constraint sets that result in improved trade-offs for more vulnerable clusters. We first introduce several notations and definitions.

Let $\bar{v}^i_{\hat{\ell}(i),[\text{R-N}]}$ be the mean estimated employment of family i in optimal location $\hat{\ell}(i)$, determined by solving FAIR-RARR in the risk-neutral setting, which is obtained by setting the $\delta_L^{\mathcal{C}_j}$ and $\delta_R^{\mathcal{C}_j}$ to sufficiently large values.

Definition 3 ***Total expected employment of families in cluster $\mathcal{C}_j$ under risk-neutral setting.*** *The aggregated employment over all families in cluster $\mathcal{C}_j$ when optimally placed by FAIR-RARR in the risk-neutral setting is defined by:*

$$E^{\mathcal{C}_j}_{[\text{R-N}]} = \sum_{i=1}^{|\mathcal{C}_j|} \bar{v}^i_{\hat{\ell}(i),[\text{R-N}]}\ ,\ \forall j \in \mathcal{J}.$$

As mentioned earlier we set $\delta_R^{\mathcal{C}_j}$ to sufficiently large values for all clusters, thus we define the measure of risk for families to be *the difference between the mean estimated employment at optimal placement from the minimum estimated employment at optimal placement.* Let $v_\ell^{i\underline{k}}$ be the minimum estimated employment of family i in location ℓ over all scenarios (with $\underline{k}$ representing the worst case scenario), and $v^{i\underline{k}}_{\hat{\ell}(i),[\text{R-N}]}$ be the minimum estimated employment of family i in optimal location $\hat{\ell}(i)$ determined by solving FAIR-RARR in the risk-neutral setting.

Definition 4 ***Total risk of families in cluster $\mathcal{C}_j$ under risk-neutral setting.*** *The aggregated risk over all families in cluster $\mathcal{C}_j$ under optimal placement by FAIR-RARR in the risk-neutral*

setting is defined by:

$$R^{\mathcal{C}_j}_{[\text{R-N}]} = \sum_{i=1}^{|\mathcal{C}_j|} \left(\bar{v}^i_{\ell(i),[\text{R-N}]} - v^{i\underline{k}}_{\ell(i),[\text{R-N}]} \right) \ , \ \forall j \in \mathcal{J}.$$

Definitions 3 and 4 are next used in the formation of new constraint sets. Let N_i be the size of family i.

Hierarchy on Average Employment Loss. A set of hierarchical constraints may be introduced into the FAIR-RARR model to ensure that the average employment loss in lower employable clusters is no worse than the average employment loss in higher employable clusters:

$$L^{\mathcal{C}_j} = E^{\mathcal{C}_j}_{[\text{R-N}]} - \sum_{i=1}^{|\mathcal{C}_j|} \sum_{\ell=1}^{|\mathcal{L}|} \bar{v}^i_\ell x^i_\ell \ , \ \forall j \in \mathcal{J}, \tag{3.7a}$$

$$\frac{L^{\mathcal{C}_j}}{\sum_{i=1}^{|\mathcal{C}_j|} N_i} \ \leq \ \frac{L^{C_{j+1}}}{\sum_{i=1}^{|C_{j+1}|} N_i} \ , \ \forall j \in \{0, 1, ..., |\mathcal{J}| - 2\}. \tag{3.7b}$$

Hierarchy on Average Risk Gain. A set of hierarchical constraints may be introduced into the FAIR-RARR model to ensure that the average risk gain in lower employable clusters is no worse than the average risk gain in higher employable clusters:

$$G^{\mathcal{C}_j} = R^{\mathcal{C}_j}_{[\text{R-N}]} - \sum_{i=1}^{|\mathcal{C}_j|} \sum_{\ell=1}^{|\mathcal{L}|} \left(\bar{v}^i_\ell - v^{i\underline{k}}_\ell \right) x^i_\ell \ , \ \forall j \in \mathcal{J}, \tag{3.8a}$$

$$\frac{G^{\mathcal{C}_j}}{\sum_{i=1}^{|\mathcal{C}_j|} N_i} \ \geq \ \frac{G^{C_{j+1}}}{\sum_{i=1}^{|C_{j+1}|} N_i} \ , \ \forall j \in \{0, 1, ..., |\mathcal{J}| - 2\}. \tag{3.8b}$$

As we are interested in lower employment loss and higher risk gain for lower employable clusters, we may also introduce constraints on the gain-to-loss ratio. Such constraints require normalization, as gain and loss differ in scale. We present two such normalization approaches, and for each introduce a gain-to-loss ratio and discuss how these ratios can be beneficial in designing constraints that encourage equitable outcomes for more vulnerable refugee families.

Hierarchy on Gain-to-Loss Ratio as Percentage of Risk-Neutral. The first normalized gain-to-loss ratio we propose represents the risk gain and employment loss of each cluster as a percentage of the total risk and total employment of that cluster at the *risk-neutral* setting. This gain-to-loss ratio demonstrates the performance of the cluster compared to that of the risk-neutral setting. We make the small technical assumption that the total risk and total employment for every cluster j is positive.

$$\frac{G^{\mathcal{C}_j} / R^{\mathcal{C}_j}_{[\text{R-N}]}}{L^{\mathcal{C}_j} / E^{\mathcal{C}_j}_{[\text{R-N}]}} \geq \frac{G^{\mathcal{C}_{j+1}} / R^{\mathcal{C}_{j+1}}_{[\text{R-N}]}}{L^{\mathcal{C}_{j+1}} / E^{\mathcal{C}_{j+1}}_{[\text{R-N}]}} \ , \ \forall j \in \{0, 1, ..., |\mathcal{J}| - 2\}. \tag{3.9}$$

Hierarchy on Gain-to-Loss Ratio as Percentage of Total. We propose a second normalized gain-to-loss ratio that represents the risk gain and employment loss of each cluster as a percentage of the total risk and total employment that is achieved by the *risk-averse* model. This ratio demonstrates, per cluster, the proportion of risk reduction to the total risk reduction, and the proportion of employment loss to the total employment loss.

$$L^{\mathcal{F}} = \sum_{j=0}^{|\mathcal{J}|} L^{\mathcal{C}_j} \tag{3.10a}$$

$$G^{\mathcal{F}} = \sum_{j=0}^{|\mathcal{J}|} G^{\mathcal{C}_j} \tag{3.10b}$$

$$\frac{G^{\mathcal{C}_j}/G^{\mathcal{F}}}{L^{\mathcal{C}_j}/L^{\mathcal{F}}} \geq \frac{G^{\mathcal{C}_{j+1}}/G^{\mathcal{F}}}{L^{\mathcal{C}_{j+1}}/L^{\mathcal{F}}}\ ,\ \forall j \in \{0, 1, ..., |\mathcal{J}| - 2\}. \tag{3.10c}$$

While nonlinear expressions 3.9 and 3.10c may be linearizable, which we leave for future work. Further, more than one of these constraint sets can be used simultaneously.

3.5.3 Post-Optimality Analysis

Augmenting Fair-RARR models employing strong $\delta_L^{\mathcal{C}_j}$ and $\delta_R^{\mathcal{C}_j}$ values with any of the aforementioned constraint sets can eliminate many undesirable solutions. Even so, multiple parameter sets may still lead to respective optimal solutions that are desirable according to expert judgement. Post-optimality analysis may be used to further explore these solutions to determine one that is most fitting. In particular, interactive visualization can guide decision makers to observe resulting solutions and evaluate their respective performance.

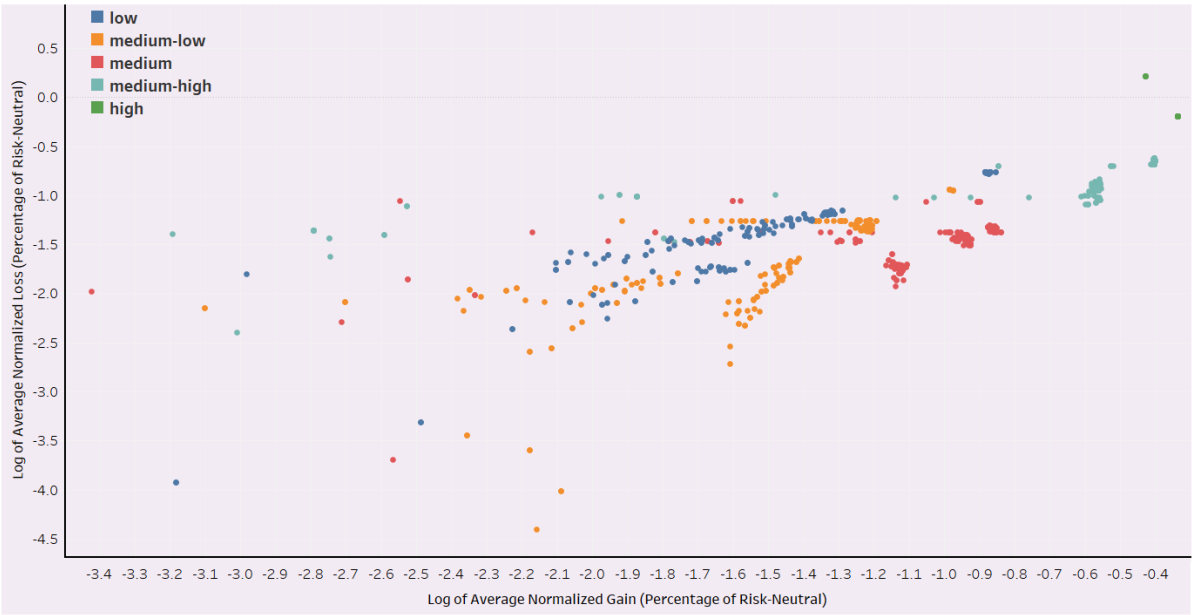
Figure 3.6 depicts an illustrative subset of resulting solutions from which a decision maker may seek to choose. Circles in the graph are color-coded and correspond to clusters. The vertical axis shows the average loss on employment (percentage of risk-neutral), formulated as:

$$\frac{G^{\mathcal{C}_j}/R^{\mathcal{C}_j}_{[\text{R-N}]}}{\sum_{i=1}^{|\mathcal{C}_j|} N_i}$$

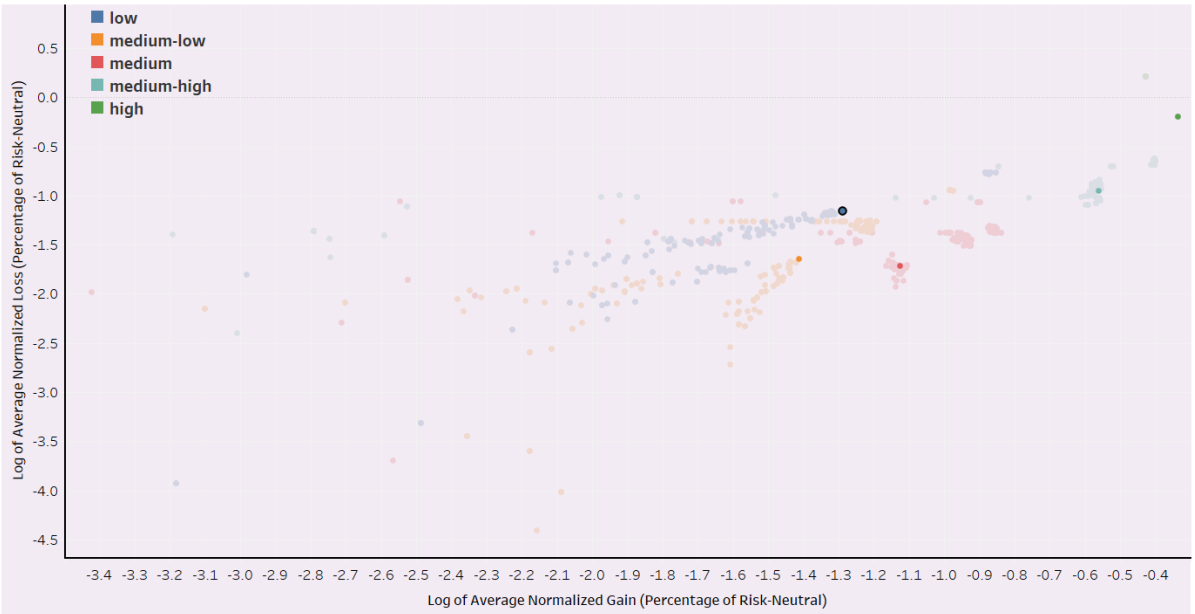
The horizontal axis shows the average gain on risk (percentage of risk-neutral), formulated as:

$$\frac{L^{\mathcal{C}_j}/E^{\mathcal{C}_j}_{[\text{R-N}]}}{\sum_{i=1}^{|\mathcal{C}_j|} N_i}$$

Hovering on any circle causes all other unassociated circles to fade away, leaving only five circles in the forefront. Each of the five remaining circles visually represent the performance of a single optimal solution resulting from a single set of parameters. In this way, a best solution can be visually pursued according to the performance on two key metrics for all clusters of every solution.



(a) Interactive visualization; circles represent the performance of different resulting optimal solutions. Different colored circles correspond to different clusters.



(b) Interactive visualization; hovering on any circle, only the circles pertaining to the performance of a single optimal solution (a set of parameter values) can be seen at the same time.

Figure 3.6: Interactive visualization to observe and evaluate the performance of different optimal solutions generated by different parameter settings.

3.6 Conclusion and Future Work

We consider risk in the context of placement optimization for refugee resettlement. Several papers to date have considered optimization from the perspective of optimizing expected employment [61, 75, 93], as well as preferences [93–95], but to the best of our knowledge, none have considered risk. Accordingly, we propose several optimization models that determine optimal placements for refugee families while considering the risk associated with their placements, specifically, uncertainty in estimated employment outcomes. We explain how risk broadly differs for vulnerable people, here in the context of refugee resettlement, from the classical notion of risk in the optimization literature. Instead of the classical perspective that centers on what decision makers gain, we rather focus on the uncertainty around what each refugee family gains. Incorporating this alternative family-level definition of risk better addresses the needs of vulnerable refugee families.

We further introduce the notion of fairness in risk-averse refugee resettlement, proposing a novel mathematical optimization formulation that hedges against risk, Fair-RARR. We demonstrate how Fair-RARR is able to assign relatively less risk to the most vulnerable of refugees. We develop unsupervised learning methods, informed by supervised learning outcomes (namely employment likelihoods), to cluster refugees, and demonstrate how these clusters can be used in our fair optimization modeling to redistribute relatively greater refugee-level risk to the those refugees exhibiting the least vulnerability.

While our Fair-RARR model gives freedom to decision makers to adjust the level of risk reduction for different clusters of families by adjusting risk tolerance parameters, further investigation on their appropriate selection is warranted to aid decision makers in obtaining insights on the dynamics of the employment-risk trade-off over clusters of families, and how risk can be distributed over different clusters by choosing different sets of parameters. Our optimization models that address individual risk are computationally efficient as compared with the classical risk based model, Collective-RARR. In the future, additional experiments can be conducted to study the computational efficiency of these models on larger data sets.

Conclusions

This dissertation explores the theory and application of analytics for refugee resettlement. The algorithmic solutions proposed in this dissertation have broad practical implications that can benefit the United States and other resettling countries, particularly in the country's efforts to optimally resettle refugees in a manner that maximize their current and future welfare, improve their ability to find gainful employment, provide for their families, and assist with successful integration. The United Nations High Commissioner for Refugees (UNHCR) notes that resettlement is vital for protecting the most vulnerable of refugees, helping to save lives and to ensure that those who are at the greatest risk are provided with options to live fulfilling lives in their host countries. The United States has historically served as one of the largest refugee resettlement countries in the world, with 30,000 refugees resettled in the country in fiscal year 2019. To ensure that refugees are resettled in areas of the United States in which they are provided with the greatest opportunities to support themselves and to contribute to their local communities, it is vital that such research be maintained.

In Chapter 1, we use machine learning to train predictive models on past refugee placement and outcome data to estimate employment likelihoods of refugees in communities. These estimated values are then used as refugee-community match quality scores in an integer optimization model for optimal matching of to-be-arriving refugees into the network of communities. We leveraged interactive visualization to implement our algorithmic solution into the design and development of the world's first refugee resettlement decision support software, *Annie*™ Moore (Matching and Outcome Optimization for Refugee Empowerment). The purpose of this software is to improve the process by which resettlement agencies arrive upon a final assignment of refugees to local communities. The first version of *Annie*™ was deployed at US refugee resettlement agency HIAS in May 2018 with new version updates over time. Our software overcomes several inefficiencies of conventional resettlement, and computational testing shows that it improves expected integration outcomes for refugees and communities.

In Chapter 2, we built upon our previous developments in refugee resettlement using data analytics, focusing on the dynamic nature of the resettlement process. To improve refugee employment outcomes in light of this dynamic nature, we adapt our optimization model to a more dynamic optimization framework that is able to carefully weigh placement decisions considering current capacity and time parameters. Our dynamic algorithms are able to calculate the value of units of community resettlement capacity and optimize placement in light of expected employment of refugee families in communities and total remaining capacity in every community. Our methods obtain improved yearly employment outcomes, demonstrated on data from 2014 to 2019. We implemented the results of our enhanced model in the refugee

resettlement decision support software *Annie*™, deployed for HIAS.

In Chapter 3, we extended the optimization model from Chapter 3 to account for risk in placement optimization for refugee resettlement. The inherit uncertainty that exists with respect to the estimation of refugee families outcomes in different locations, results in risk in the placement optimization of refugee families. We developed two mathematical formulations for risk-averse optimization models that place families in less-risky locations while maximizing their expected outcomes. Both our models are attuned to *family-level risk* that accounts for the vulnerability of refugee families. Our models interpret different measures for family-level risk. While our first model hedges against the risk in the objective function and is computationally efficient, a second model that limits the risk in the constraint space is able to address fairness in refugee resettlement through a scheme that assigns relatively less risk to those refugees that are the most vulnerable. We show that risk-averse optimization models can alleviate much of the risk while retaining much of the total expected employment.

References

[1] UNHCR. UNHCR - Figures at a Glance. https://www.unhcr.org/en-us/figures-at-a-glance.html, 2021.

[2] UNHCR. Global trends: Forced displacement in 2020. Technical report, United Nations High Commissioner for Refugees, 2021.

[3] Ozlem Karsu, Bahar Y. Kara, and Bayram Selvi. The refugee camp management: a general framework and a unifying decision-making model. *Journal of Humanitarian Logistics and Supply Chain Management*, 2019.

[4] UNHCR. Global Resettlement Needs 2022. Technical report, United Nations High Commissioner for Refugees, June 2021.

[5] UNHCR. Global Resettlement Needs 2021. Technical report, United Nations High Commissioner for Refugees, June 2020.

[6] Andrew C. Trapp. Refugee resettlement via analytics. https://pubsonline.informs.org/do/10.1287/orms.2019.02.09/full/, Mar 2019.

[7] Olof Åslund and Dan-Olof Rooth. Do when and where matter? Initial labour market conditions and immigrant earnings. *The Economic Journal*, 117(518):422–448, 2007.

[8] Olof Åslund and Peter Fredriksson. Peer effects in welfare dependence quasi-experimental evidence. *Journal of Human Resources*, 44(3):798–825, 2009.

[9] Olof Åslund, John Östh, and Yves Zenou. How important is access to jobs? Old question, improved answer. *Journal of Economic Geography*, 10(3):389–422, 2010.

[10] Olof Åslund, Per-Anders Edin, Peter Fredriksson, and Hans Grönqvist. Peers, neighborhoods, and immigrant student achievement: Evidence from a placement policy. *American Economic Journal: Applied Economics*, 3(2):67–95, 2011.

[11] Anna Piil Damm. Neighborhood quality and labor market outcomes: Evidence from quasi-random neighborhood assignment of immigrants. *Journal of Urban Economics*, 79: 139–166, 2014.

[12] Jeremy Ferwerda and Justin Gest. Location, location: Refugee resettlement and employment outcomes in the United States. Technical report, Dartmouth College, 2017.

[13] Jesús Fernández-Huertas Moraga and Hillel Rapoport. Tradable immigration quotas. *Journal of Public Economics*, 115:94–108, 2014.

[14] Will Jones and Alexander Teytelboym. The local refugee match: Aligning refugees' preferences with the capacities and priorities of localities. *Journal of Refugee Studies*, 31: 152–178, 2018.

[15] Tommy Andersson and Lars Ehlers. Assigning refugees to landlords in Sweden: Efficient, stable, and maximum matchings. *The Scandinavian Journal of Economics*, 122(3):937–965, 2017.

[16] Haris Aziz, Jiayin Chen, Serge Gaspers, and Zhaohong Sun. Stability and Pareto optimality in refugee allocation matchings. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 964–972. International Foundation for Autonomous Agents and Multiagent Systems, 2018.

[17] Alvin E. Roth. Marketplaces, markets, and market design. *American Economic Review*, 108(7):1609–58, 2018.

[18] David Delacrétaz, Scott Duke Kominers, and Alexander Teytelboym. Matching mechanisms for refugee resettlement. Technical report, University of Oxford, 2019.

[19] Thành Nguyen, Hai Nguyen, and Alexander Teytelboym. Stability in matching markets with complex constraints. In *Proceedings of the 2019 ACM Conference on Economics and Computation*, pages 61–61. ACM, 2019.

[20] Kirk Bansak, Jeremy Ferwerda, Jens Hainmueller, Andrea Dillon, Dominik Hangartner, Duncan Lawrence, and Jeremy Weinstein. Improving refugee integration through data-driven algorithmic assignment. *Science*, 359(6373):325–329, 2018.

[21] UNHCR. Global trends: Forced displacement in 2018. Technical report, United Nations High Commissioner for Refugees, Geneva, 2019.

[22] UNHCR. Global Resettlement Needs 2020. Technical report, United Nations High Commissioner for Refugees, July 2019.

[23] Will Jones and Alexander Teytelboym. The international refugee match: A system that respects refugees' preferences and the priorities of states. *Refugee Survey Quarterly*, 36 (2):84–109, 2017.

[24] U.S. Department of State. Refugee Arrivals by State and Nationality as of Mar. 6, 2020. `http://www.wrapsnet.org/admissions-and-arrivals`, 2020. Accessed: 2020-03-18.

[25] Alastair Ager and Alison Strang. Understanding integration: A conceptual framework. *Journal of Refugee Studies*, 21(2):166–191, 2008.

[26] Gary Lichtenstein, Jini Puma, Amy Engelman, and Maggie Miller. The refugee integration survey evaluation project (RISE) year five: Final report, 2016.

[27] Vincent W. Slaugh, Mustafa Akan, Onur Kesten, and M. Utku Ünver. The Pennsylvania Adoption Exchange improves its matching process. *Interfaces*, 46(2):133–153, 2016.

[28] Andrew C. Trapp, Alexander Teytelboym, Narges Ahani, and Tommy Andersson. Refugee resettlement via machine learning and integer optimization. In *OR60 Annual Conference – Keynote Papers and Extended Abstracts*. The OR Society, 2018.

[29] Alfonso J. Pedraza-Martinez and Luk N. Van Wassenhove. Empirically grounded research in humanitarian operations management: The way forward. *Journal of Operations Management*, 45:1–10, 2016.

[30] Maria Besiou, Alfonso J. Pedraza-Martinez, and Luk N. Van Wassenhove. OR applied to humanitarian operations. *European Journal of Operational Research*, 269:397–405, 2018.

[31] Maria Besiou and Luk N. Van Wassenhove. Humanitarian operations: A world of opportunity for relevant and impactful research. *Manufacturing & Service Operations Management*, 22(1):135–145, 2020.

[32] Wenwei Cao, Melih Çelik, Özlem Ergun, Julie Swann, and Nadia Viljoen. Challenges in service network expansion: An application in donated breastmilk banking in South Africa. *Socio-Economic Planning Sciences*, 53:33–48, 2016.

[33] Melih Çelik, Özlem Ergun, Ben Johnson, Pınar Keskinocak, Álvaro Lorca, Pelin Pekgün, and Julie Swann. Humanitarian logistics. In *New Directions in Informatics, Optimization, Logistics, and Production*, pages 18–49. INFORMS, 2012.

[34] Silvano Martello and Paolo Toth. Solution of the zero-one multiple knapsack problem. *European Journal of Operational Research*, 4(4):276–283, 1980.

[35] Arnaud Fréville. The multidimensional 0–1 knapsack problem: An overview. *European Journal of Operational Research*, 155(1):1–21, 2004.

[36] Yang Song, Chi Zhang, and Yuguang Fang. Multiple multidimensional knapsack problem and its applications in cognitive radio networks. In *MILCOM 2008-2008 IEEE Military Communications Conference*, pages 1–7. IEEE, 2008.

[37] Mhand Hifi, Mustapha Michrafy, and Abdelkader Sbihi. Heuristic algorithms for the multiple-choice multidimensional knapsack problem. *Journal of the Operational Research Society*, 55(12):1323–1332, 2004.

[38] Per-Anders Edin, Peter Fredriksson, and Olof Åslund. Ethnic enclaves and the economic success of immigrants—evidence from a natural experiment. *Quarterly Journal of Economics*, 118(1):329–357, February 2003.

[39] Eleonora Patacchini and Yves Zenou. Ethnic networks and employment outcomes. *Regional Science and Urban Economics*, 42(6):938–949, November 2012.

[40] Gurobi. *Gurobi Optimizer 9.0.0 Reference Manual*. Gurobi Optimization, Inc., 2020.

[41] Stuart Mitchell, Michael O'Sullivan, and Iain Dunning. PuLP: A linear programming toolkit for Python. `https://github.com/coin-or/pulp`, 2020.

[42] COIN-OR. CBC user guide. `http://www.coin-or.org/cbc`, 2020.

[43] Armin Ronacher. Flask. `https://palletsprojects.com/p/flask/`, 2020.

[44] David Meignan, Sigrid Knust, Jean-Marc Frayret, Gilles Pesant, and Nicolas Gaud. A review and taxonomy of interactive optimization methods in operations research. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 5(3):1–43, 2015.

[45] Tommy Andersson, Lars Ehlers, and Alessandro Martinello. Dynamic refugee matching. Working paper, Lund University, 2018.

[46] Stephen Castles, Maja Korac, Ellie Vasta, and Steven Vertovec. Integration: Mapping the field. Technical report, Centre for Migration and Policy Research and Refugee Studies Centre, University of Oxford, 2002.

[47] Michael P. Johnson, Gerald Midgley, and George Chichirau. Emerging trends and new frontiers in community operational research. *European Journal of Operational Research*, 268:1178–1191, 2018.

[48] Nils Olberg and Sven Seuken. Enabling trade-offs in machine learning-based matching for refugee resettlement. Technical report, University of Zurich, 2019.

[49] Péter Biró and Jens Gudmundsson. Complexity of finding Pareto-efficient allocations of highest welfare. *European Journal of Operational Research*, 2020.

[50] Alvin E. Roth. *Who Gets What - And Why: Understand the Choices You Have, Improve the Choices You Make*. William Collins, 2016.

[51] Atila Abdulkadiroğlu and Tayfun Sönmez. School choice: A mechanism design approach. *American Economic Review*, 93:729–747, 2003.

[52] Alvin E. Roth, Tayfun Sönmez, and M. Utku Ünver. Kidney exchange. *Quarterly Journal of Economics*, 119:457–488, 2004.

[53] José Alcalde and Salvador Barberà. Top dominance and the possibility of strategy-proof stable solutions to the marriage problem. *Economic Theory*, 4:417–435, 1994.

[54] Salvador Barberà and Matthew Jackson. Strategy-proof exchange. *Econometrica*, 63:51–87, 1995.

[55] Alvin E. Roth. The economics of matching: Stability and incentives. *Mathematics of Operations Research*, 7:617–628, 1982.

[56] Tayfun Sönmez. Strategy-proofness and essentially singled-valued cores. *Econometrica*, 67:677–689, 1999.

[57] David Bolt. An inspection of the vulnerable persons resettlement scheme. Technical report, Independent Chief Inspector of Borders and Immigration, 2018.

[58] SOU. Ett ordnat mottagande – gemensamt ansvar för etablering eller återvändande. Statens Offentliga Utredningar, SOU 2018:22,, Stockholm, 2018.

[59] UNHCR. Global trends: Forced displacement in 2019. Technical report, United Nations High Commissioner for Refugees, 2020.

[60] Linna Martén, Jens Hainmueller, and Dominik Hangartner. Ethnic networks can foster the economic integration of refugees. *Proceedings of the National Academy of Sciences*, 116(33):16280–16285, 2019.

[61] Narges Ahani, Tommy Andersson, Alessandro Martinello, Alexander Teytelboym, and Andrew C. Trapp. Placement optimization in refugee resettlement. *Operations Research*, 2021.

[62] Dilek Sayedahmed. Centralized refugee matching mechanisms with hierarchical priority classes. Mimeo, 2020.

[63] Gian Caspari. An alternative approach to asylum assignment. Mimeo, November 2019.

[64] Kirk Bansak. A minimum-risk dynamic assignment mechanism along with an approximation, heuristics, and extension from single to batch assignments. *arXiv preprint arXiv:2007.03069*, 2020.

[65] Jon Feldman, Nitish Korula, Vahab Mirrokni, Shanmugavelayutham Muthukrishnan, and Martin Pál. Online Ad assignment with free disposal. In S. Leonardi, editor, *Proceedings of the 5th International Workshop on Internet and Network Economics*, pages 374–385, 2009. ISBN 978-3-642-10841-9.

[66] Thomas Kesselheim, Klaus Radke, Andreas Tönnis, and Berthold Vöcking. An optimal online algorithm for weighted bipartite matching and extensions to combinatorial auctions. In *Proceedings of the 21st Annual European Symposium on Algorithms*, pages 589–600, 2013.

[67] Saeed Alaei, MohammadTaghi Hajiaghayi, and Vahid Liaghat. The online stochastic generalized assignment problem. In *Proceedings of the 16th International Workshop on Randomization and Approximation Techniques in Computer Science*, pages 11–25. 2013.

[68] John Dickerson, Karthik Sankararaman, Kanthi Sarpatwar, Aravind Srinivasan, Kung-Lu Wu, and Pan Xu. Online resource allocation with matching constraints. In *Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems*, 2019.

[69] Nikhil R. Devanur and Thomas P. Hayes. The adwords problem: Online keyword matching with budgeted bidders under random permutations. In *Proceedings of the 10th ACM Conference on Electronic Commerce*, pages 71–78, 2009.

[70] Erik Vee, Sergei Vassilvitskii, and Jayavel Shanmugasundaram. Optimal online assignment with forecasts. In *Proceedings of the 11th ACM Conference on Electronic Commerce*, pages 109–118, 2010. ISBN 978-1-60558-822-3.

[71] John Dickerson, Ariel Procaccia, and Tuomas Sandholm. Dynamic matching via weighted myopia with application to kidney exchange. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 26, 2012.

[72] John Dickerson and Tuomas Sandholm. Futurematch: Combining human value judgments and machine learning to match in dynamic environments. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

[73] Irving Jones. Home Away From Home. *State Magazine*, pages 17–20, December 2015.

[74] Herman B. Leonard. Elicitation of honest preferences for the assignment of individuals to positions. *Journal of Political Economy*, 91(3):461–479, 1983.

[75] Narges Ahani, Paul Gölz, Ariel D Procaccia, Alexander Teytelboym, and Andrew C Trapp. Dynamic placement in refugee resettlement. *arXiv preprint arXiv:2105.14388*, 2021.

[76] Yasemin Merzifonluoglu and Joseph Geunes. The risk-averse static stochastic knapsack problem. *INFORMS Journal on Computing*, 2020.

[77] Chao Ning and Fengqi You. Optimization under uncertainty in the era of big data and deep learning: When machine learning meets mathematical programming. *Computers & Chemical Engineering*, 125:434–448, 2019.

[78] Evelyn ML. Beale. On minimizing a convex function subject to linear inequalities. *Journal of the Royal Statistical Society: Series B (Methodological)*, 17(2):173–184, 1955.

[79] George B. Dantzig. Linear programming under uncertainty. *Management Science*, 50 (12_supplement):1764–1769, 2004.

[80] Roger J-B. Wets. Programming under uncertainty: the equivalent convex program. *SIAM Journal on Applied Mathematics*, 14(1):89–105, 1966.

[81] John R. Birge and Francois Louveaux. *Introduction to stochastic programming*. Springer Science & Business Media, 2011.

[82] Abraham Charnes and William W. Cooper. Chance-constrained programming. *Management science*, 6(1):73–79, 1959.

[83] David H Pyle and Stephen J Turnovsky. Risk aversion in chance constrained portfolio selection. *Management Science*, 18(3):218–225, 1971.

[84] Andras Prékopa. Contributions to the theory of stochastic programming. *Mathematical Programming*, 4(1):202–221, 1973.

[85] Abraham Wald. Statistical decision functions which minimize the maximum risk. *Annals of Mathematics*, pages 265–280, 1945.

[86] Aharon Ben-Tal and Arkadi Nemirovski. Robust solutions of uncertain linear programs. *Operations research letters*, 25(1):1–13, 1999.

[87] Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. Robust optimization. In *Robust optimization*. Princeton university press, 2009.

[88] Dimitris Bertsimas and Melvyn Sim. The price of robustness. *Operations research*, 52(1): 35–53, 2004.

[89] R Tyrrell Rockafellar, Stanislav Uryasev, et al. Optimization of conditional value-at-risk. *Journal of risk*, 2:21–42, 2000.

[90] Andrzej Ruszczyński and Alexander Shapiro. Optimization of risk measures. *Probabilistic and randomized methods for design under uncertainty*, pages 119–157, 2006.

[91] Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczyński. *Lectures on stochastic programming: modeling and theory*. SIAM, 2014.

[92] Harry M. Markowitz. *Portfolio selection*. Yale university press, 1968.

[93] Avidit Acharya, Kirk Bansak, and Jens Hainmueller. Combining outcome-based and preference-based matching: A constrained priority mechanism. *Political Analysis*, 30 (1):89–112, 2022.

[94] Will Jones and Alexander Teytelboym. The local refugee match: Aligning refugees' preferences with the capacities and priorities of localities. *Journal of Refugee Studies*, 31(2): 152–178, 2018.

[95] Tommy Andersson, Lars Ehlers, and Alessandro Martinello. Dynamic refugee matching. *Cahier de recherche*, (2018-10), 2018.

[96] Justin Hsu, Jamie Morgenstern, Ryan Rogers, Aaron Roth, and Rakesh Vohra. Do prices coordinate markets? In *Proceedings of the 48th Annual ACM Symposium on Theory of Computing*, pages 440–453, 2016.

Appendices

Appendix A

Chapter 1: Data Appendix

We obtain anonymized data on all individual refugees relocated by HIAS between 2010 and 2017. We focus on free cases, that is, refugees that can be freely allocated across affiliates as they have no pre-existing family ties. As stated in the main text, we use refugees arriving until 2016 to train our models, and those arriving in 2017 as a test sample. Note that the quota-relevant year starts on October 1. Therefore, 2017 refugees are those arriving from October 1 2016 to September 30 2017. After the split, we observe 2,486 refugees in the training sample and 498 refugees in the test sample.

What follows is a list of data features and definitions.

- Arrival Date: The years span 2010 through 2017, inclusive.
- Case Number: This is an anonymized, unique identifier for each family; in total, there are 1,896 families and 5,326 refugees.
- Relationship Code: The relationship to the principal applicant for each individual in a family; these include Principal Applicant (PA), Husband (HU), Wife (WI), Daughter (DA), Son (SO), Stepdaughter (SD), Stepson (SN).
- Gender Code: Genders include Male and Female.
- Nationality: There are 33 nationalities represented.
- Language: There are 133 languages represented, with proficiency levels for reading, speaking, and writing.
- Education Level: Levels include kindergarten, primary, intermediate, secondary, technical school, pre-university, university, professional, and graduate school.
- Medical Condition: There are at least 31 types of medical conditions.
- Treatment Urgency: There are several levels indicating the degree of treatment urgency, including Ongoing, Immediate, Urgent.
- Urgency Code: This is how fast the case must be assured by the resettlement agency. Values include both normal and expedited (such as medical, protection, etc.).
- Affiliate: This is the local community to which family is resettled.

- Employed: This is a binary value indicating whether the refugee was employed 90 days after arrival.
- Age Upon Arrival

Summary statistics for the above features include:

- Average case size: The average size differs among nationalities, affiliates, and year of arrival. Across all cases, the average size is approximately 2.809.
- Average age: The average age is approximately 23 years; 42.81% of refugees are under the age of 18, 55.97% are between 18 to 64, and 1.22% are beyond 64 years of age.
- Total number of nationalities: The refugees originate from 33 different nationalities; 96% of which derive from 13 countries.
- Total number of languages: There are 133 different languages among all refugees.
- Fraction with tertiary education: 6.04% of all refugees (10.57% of adult refugees) have a tertiary education.

To estimate counterfactual employment probabilities (Section 1.4 of the paper), we recode and transform some of the observed features. From Relationship Code we create an indicator of being a single parents, and a counter (censored at 5) of the number of children in the household. From Language we obtain an indicator for English speaking and a counter of the number of languages spoken. From Medical Condition we create an indicator for whether the refugee suffers from any medical condition, and a counter (censored at 5) of the total number of medical conditions reported. We recode Education Level into four groups (less than secondary schooling, secondary schooling, advanced—but not college—degrees, and university and college level degrees). Finally, we use the primary Nationality to group refugees in their area of origin (Africa, Middle East, Asia, or Other; note that we classify Oman, Lebanon, Iraq, Yemen, Iran, Bahrain, Syria, Qatar, Jordan, Kuwait, Israel, U.A.E. and Saudi Arabia as Middle East rather than Asia to better differentiate refugees from the Arabian peninsula and those from East Asia). For estimating LASSO, we also manually construct interactions between these variables and add a second order polynomial in age. The full list of features used in the LASSO and GBRT models appears in Tables B.1 and B.2.

To correctly account for changes in the average level of employment over time, we add to the data quarter-specific macro-economic variables, that is, average US employment level (adjusted for seasonality) and average unemployment rate (not adjusted by seasonality). Note that we add non-adjusted unemployment rates to capture seasonality in employment probabilities; whether we adjust employment ratios or unemployment rates for seasonality does not matter for our predictions. In the interacted logit and LASSO models these macro variables do not interact with affiliates, as their purpose is simply to adjust the varying average employment level of refugees over time.

Appendix B

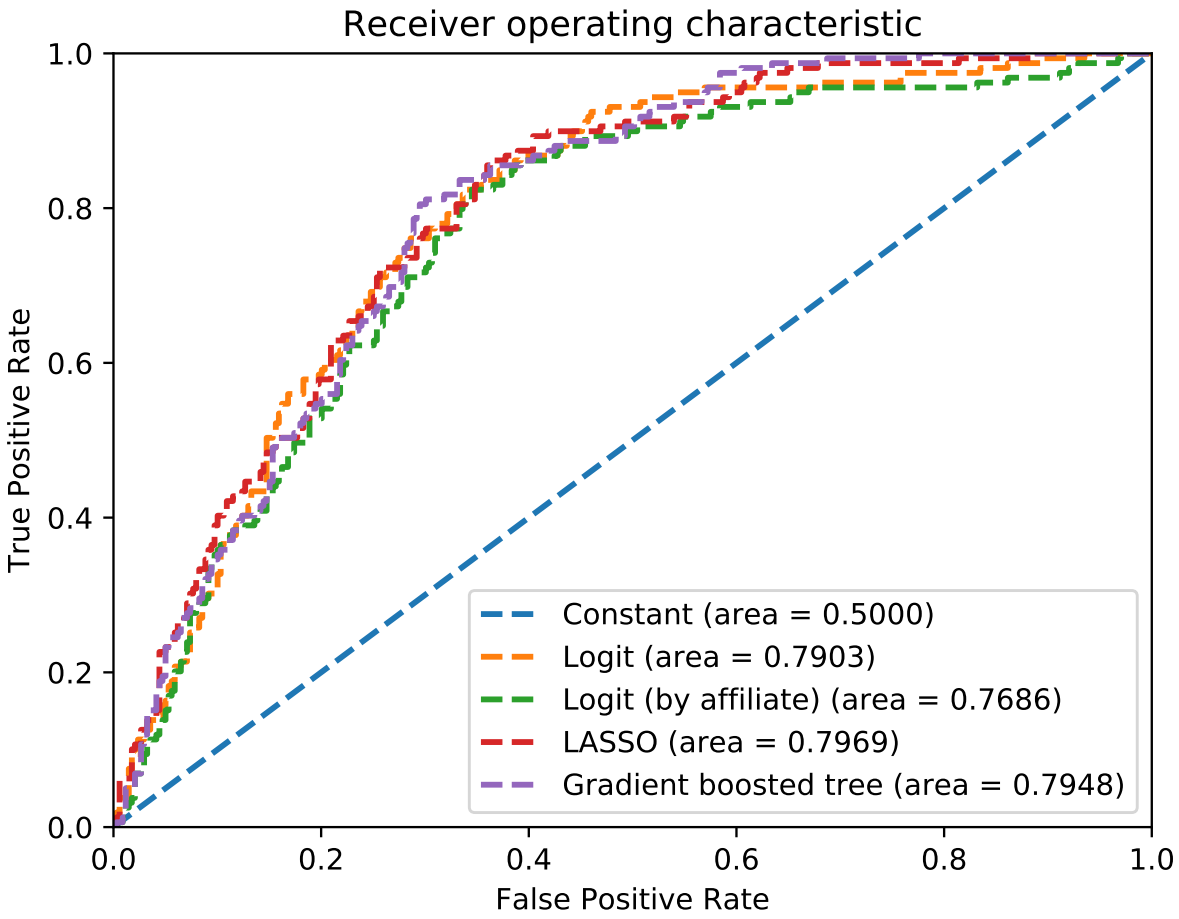
Chapter 1: Machine Learning Models: Procedure and Diagnostics

As stated in the main text, we restrict our data to refugees arriving between 2010 and 2016 for training our models, and test them on data for refugees arriving in 2017. For LASSO, we build a series of feature interactions, and then again fully interact this data matrix for each of the seven affiliates receiving at least 200 refugees until 2016. We standardize each feature such that it ranges from 0 to 1 in the training data (we use maxima and minima of the training set to standardize the test set). We use 5-fold cross-validation targeting the in-sample area under the Receiver Operating Characteristic (ROC) curve to tune model hyper-parameters.

Figure B.1 shows Receiver Operating Characteristic (ROC) curves for LASSO, GBRT, and all benchmark models in the test data. ROC curves plot the achievable fraction of true positives as a function of the admissible false positives. The higher the fraction of true positives achievable for a given fraction of false positive is, the better is the performance of the model. Thus, curves to the northwest of the graph dominate the others. The graph shows that both LASSO and GBRT produce higher AUC-ROC than the benchmark models.

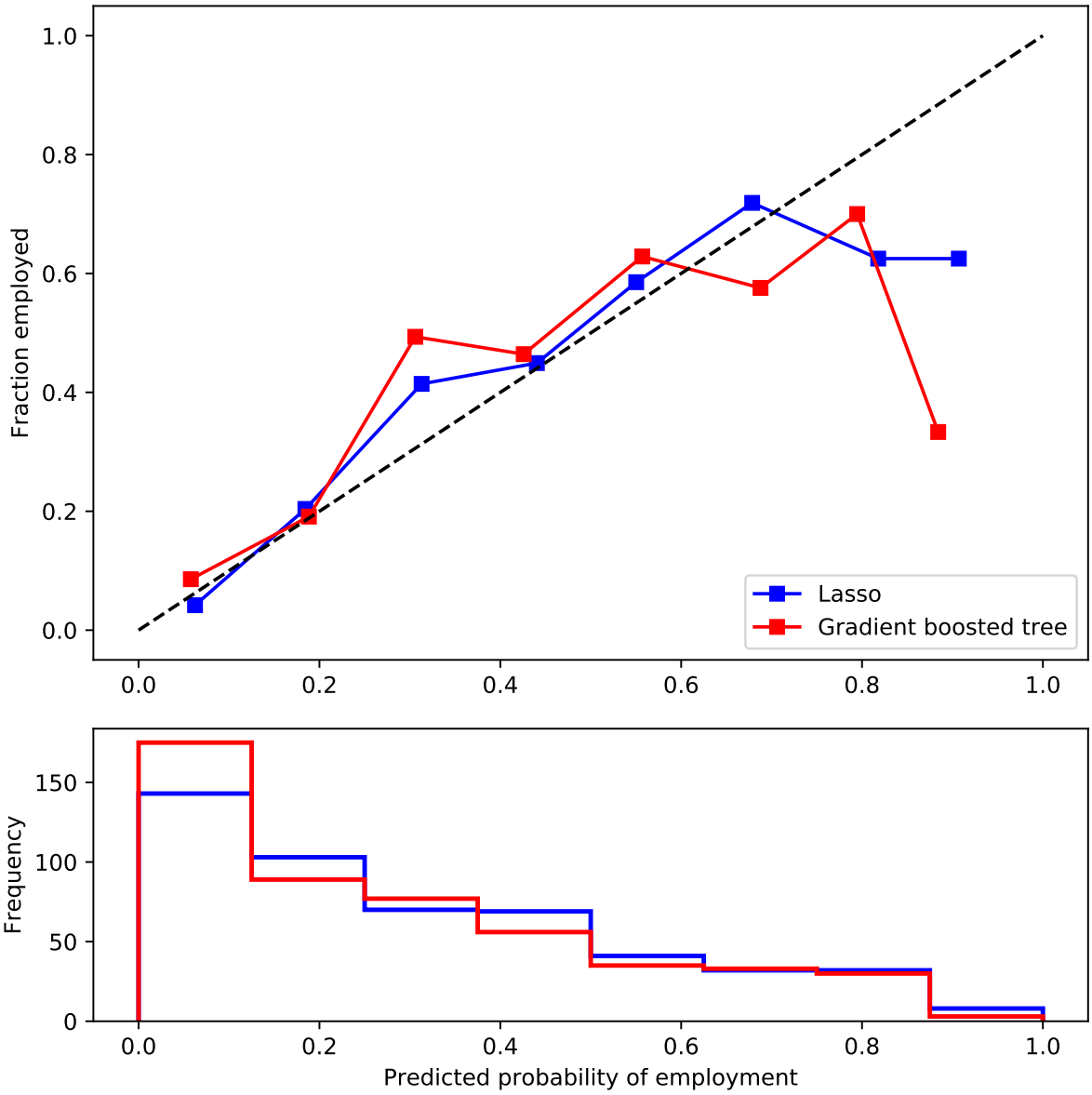
For both the GBRT and LASSO models, Figure B.2 also shows calibration plots, depicting the average number of employed refugees in the test set for given predicted probabilities of employment. It is apparent that the predicted probabilities of employment after 90 days can be high for refugees and range from zero to approximately 0.8. This range of predicted probabilities for the US is in stark contrast with that observable in Europe, where predicted probabilities of employment rarely exceed 0.5 [20]. LASSO is well calibrated up to very high predicted probabilities, for which in the test data we observe a lower rate of employment than predicted. This behavior is primarily due to our out-of-sample extrapolation using macro-economic data for the affiliates for which we have little data. Without the inclusion of macro data as model features both LASSO and GBRT models are better calibrated, but tend to under-predict average employment levels.

The remainder of this Appendix reports normalized feature (Gini) importance scores for GBRT and model coefficients for LASSO. Note that these scores and coefficients, while broadly indicative of the amount of explanatory power contained in each feature, should not be taken as direct measures of feature relevance, especially as most features in our data are strongly correlated with one another. This point is particularly relevant for LASSO coefficients. While we standardize all model features such that they range from zero to one in the training sample, their standard deviation varies considerably.



NOTE: The figure plots the fraction of achievable true positives as a function of the fraction of false positives for each estimated model. The constant model is the benchmark used by [20]. The logit model uses the same features used in LASSO for predicting employment, but without a LASSO constraint and affiliate-specific interactions. The logit by affiliate model uses the same features used in the LASSO model (including affiliate-specific interactions), but without a LASSO constraint. We compute all functions on refugees arriving in 2017 (test sample).

Figure B.1: Receiver Operating Characteristics (ROC) curves.



Note: Both panels in the figure plot predicted employment probabilities by either LASSO or GBRT in the x-axis for the test data (refugees arriving in 2017). The top panel of the figure plots for each predicted employment probability the average number of effectively employed refugees in 2017. The bottom panel shows the histogram of the predicted employment probabilities in the test sample.

Figure B.2: Calibration plots of LASSO and GBRT models (FY17 data).

	Gini Importance
age	0.243
male	0.057
education level	0.065
case size	0.042
number of children	0.039
continent	0.069
affiliate	0.176
number of conditions	0.054
number of languages	0.024
English speaking	0.020
urgency code	0.010
primary applicant	0.022
unemployment rate (unadjusted)	0.131
employment ratio	0.049

NOTE: The table shows the normalized importance measure for each feature in the Gradient Boosted Regression Tree model. The coefficients sum to one. These measures are calculated as the average across all trees of mean decrease impurity scores for each node in which a given feature serves to split the data.

Table B.1: Feature importance in the Gradient Boosted Regression Tree (GBRT) model.

Moreover, LASSO constraints penalize coefficients different than zero. Given two strongly correlated features contributing similarly to the outcome predictions, a strong enough LASSO constraint will force one of the two associated coefficients to be equal to zero, and rely solely on the other feature for prediction. While this selection often improves the predictive performance of the model by reducing model complexity, it does not imply that the feature whose coefficient was pushed to zero has no predictive power at all. This selection simply implies that the information carried in that feature could be expressed as a function of other features without a strong loss in the in-sample explanatory power.

	Baseline	Affiliate C	Affiliate F	Affiliate I	Affiliate K	Affiliate N	Affiliate Q	Affiliate R
age	0.277				0.923			
male	1.289			-0.055	0.288			
medical condition			-0.647		-0.505	0.037	-0.207	
case size					0.922	-0.056		
number of children	-1.966		-0.243			-0.295		
single parent	0.438				0.030	0.083	-0.832	
number of conditions	-0.417			-0.797			-0.416	
number of languages	0.153				0.007			
English speaking	0.225	0.183	0.108	0.114	0.208		0.130	-0.032
urgency code		-0.334		0.715	-0.276			
age2	-2.050					-0.103		
primary applicant	0.136	-0.147		-0.021	0.637	0.040		0.642
education level_1-less than secondary				0.134		0.125		
education level_2-secondary	0.051	-0.318		-0.016				0.435
education level_3-advanced	0.719							
education level_4-university			1.003			0.680		
continent_Asia	-0.140							
continent_Middle east	-0.639							
continent_other	-0.400	1.236			0.278		1.424	0.613
1.education level#1.male	0.004	0.390						0.568
2.education level#1.male	0.120		0.203			0.022		
3.education level#1.male					-0.195			
4.education level#1.male	0.017							-0.001
c.number of children#1.male	1.134			-0.218		-0.122		
c.age#1.male								
1.primary applicant#1.male	-0.076			-0.367				
1.single parent#1.male	-0.655	1.105						
c.number of conditions#1.male		0.096		-0.220		0.213		
unemployment rate (unadjusted)								
employment ratio	0.939							
constant	-0.974	-0.289	1.805	0.713	0.208			0.373

NOTE: The table shows the estimated nonzero coefficients in the LASSO model. The first column shows the baseline coefficients of the model, while the other columns show the estimated interactions with each of the seven affiliates for which we observe at least 200 refugees before 2017.

Table B.2: Estimated coefficients in the LASSO model.

Appendix C

Chapter 1: Counterfactual Optimization Outcomes for GBRT

The outcomes from optimizing the SUM objective: $v_\ell^i = \sum_{j=1}^{|F^i|} q_\ell^{ij}$ using estimates $\hat{q}$ from the GBRT model are detailed in Table C.1, which is organized in the same manner as Table 1.2. The baseline employment levels when considering the actual (manual) placements of cases to affiliates in the data is 142.96 when using the GBRT model.

Capacity Adjustment	Min Avg Case Size	Binary Service Constraints	Total Expected Employed Refugees	Gains wrt to Predicted Employed Refugees (143)	St Dev in Avg Case Size Across Affiliates	# of Unplaced Cases / Refugees	#of Affiliates Violating 90% Capacity	# and % of Cases/Refugees Violating Constraints	Build and Run Time (s)
Observed	None	Off	175.62	22.85%	1.47	0/0	0	59/198 (17.93%/23.60%)	0.45
Observed	None	On	174.08	21.77%	1.60	2/3	0	0/0 (0.00%/0.00%)	0.41
Observed	2	Off	167.59	17.23%	0.77	1/1	0	75/195 (22.80%/23.24%)	0.94
Observed	2	On	166.21	16.26%	0.83	1/2	0	0/0 (0.00%/0.00%)	1.03
Observed	2.5	Off	159.32	11.45%	0.33	2/2	0	81/185 (24.62%/22.05%)	6.97
Observed	2.5	On	157.93	10.47%	0.14	3/4	0	0/0 (0.00%/0.00%)	7.03
Observed	3	Off	142.18	-0.55%	0.00	80/92	6	53/137 (16.11%/16.33%)	10.57
Observed	3	On	141.23	-1.21%	1.07	81/95	9	0/0 (0.00%/0.00%)	9.84
Observed	Observed	Off	161.68	13.09%	0.85	2/2	0	70/163 (21.28%/19.43%)	7.34
Observed	Observed	On	160.33	12.15%	0.84	2/3	0	0/0 (0.00%/0.00%)	6.81
$\leq 110\%$	None	Off	178.71	25.01%	1.76	0/0	3	57/179 (17.33%/21.33%)	0.85
$\leq 110\%$	None	On	177.12	23.90%	1.34	1/2	1	0/0 (0.00%/0.00%)	0.73
$\leq 110\%$	2	Off	171.96	20.28%	1.61	0/0	4	57/155 (17.33%/18.47%)	1.41
$\leq 110\%$	2	On	170.57	19.31%	1.04	1/2	1	0/0 (0.00%/0.00%)	1.25
$\leq 110\%$	2.5	Off	163.33	14.25%	0.12	0/0	1	84/197 (25.53%/23.48%)	6.25
$\leq 110\%$	2.5	On	161.93	13.27%	0.36	2/3	2	0/0 (0.00%/0.00%)	6.32
$\leq 110\%$	3	Off	145.20	1.57%	0.65	80/92	5	66/177 (20.06%/21.10%)	8.14
$\leq 110\%$	3	On	144.19	0.86%	1.07	81/95	6	0/0 (0.00%/0.00%)	6.53
$\leq 110\%$	Observed	Off	164.79	15.27%	0.99	0/0	3	80/191 (24.32%/22.77%)	11.41
$\leq 110\%$	Observed	On	163.44	14.33%	1.03	2/3	4	0/0 (0.00%/0.00%)	7.01
$[90\%, 110\%]$	None	Off	178.71	25.01%	1.27	0/0	0	53/154 (16.11%/18.36%)	0.81
$[90\%, 110\%]$	None	On	177.12	23.90%	1.22	1/2	0	0/0 (0.00%/0.00%)	1.14
$[90\%, 110\%]$	2	Off	171.96	20.28%	1.12	0/0	0	62/183 (18.84%/21.81%)	1.71
$[90\%, 110\%]$	2	On	170.57	19.31%	0.96	1/2	0	0/0 (0.00%/0.00%)	1.19
$[90\%, 110\%]$	2.5	Off	163.33	14.25%	0.32	0/0	0	72/189 (21.88%/22.53%)	4.09
$[90\%, 110\%]$	2.5	On	161.93	13.27%	0.33	2/3	0	0/0 (0.00%/0.00%)	8.83
$[90\%, 110\%]$	3	Off	Infeasible instance						
$[90\%, 110\%]$	3	On	Infeasible instance						
$[90\%, 110\%]$	Observed	Off	164.79	15.27%	0.83	4/4	0	70/181 (21.28%/21.57%)	19.75
$[90\%, 110\%]$	Observed	On	163.44	14.32%	0.86	5/6	0	0/0 (0.00%/0.00%)	13.59

Table C.1: Results of counterfactual employment optimization under various scenarios using the SUM objective and GBRT model.

Appendix D

Chapter 1: Exploration of Alternative Objective Functions

Optimizing the placement of cases to affiliates allows for multiple interpretations for translating the individual refugee-level quality scores q_ℓ^{ij} into a case-level value (or weight), v_ℓ^i for each family F^i and affiliate ℓ. While we prioritize $v_\ell^i = \sum_{j=1}^{|F^i|} q_\ell^{ij}$ (SUM) for the case-level quality score v_ℓ^i appearing in (1.1a), as mentioned in Section 1.5 other reasonable interpretations exist. These include $v_\ell^i = \max_j q_\ell^{ij}$ (MAX); $v_\ell^i = \min_j q_\ell^{ij}$ (MIN); and $v_\ell^i = 1/|N_w^i| \sum_{j \in N_w^i} q_\ell^{ij}$ (MEAN), which because they position $v_\ell^i \in [0, 1]$, are more appropriately interpreted as information about the case. We now conduct a formal study of these alternatives and explore the associated tradeoffs.

Defining $v_\ell^i = \max_j q_\ell^{ij}$ (MAX) assigns to v_ℓ^i the highest employment likelihood of the members in family F^i for affiliate L^ℓ. Correspondingly, maximizing $\sum_{i=1}^{|\mathcal{F}|} \sum_{\ell=1}^{|\mathcal{L}|} v_\ell^i z_\ell^i$ is interpreted as emphasizing placements of families into communities according to their most employable member. Alternatively, taking $v_\ell^i = \min_j q_\ell^{ij}$ (MIN) assigns to v_ℓ^i the lowest employment likelihood of the members in family F^i for affiliate ℓ. Then, maximizing $\sum_{i=1}^{|\mathcal{F}|} \sum_{\ell=1}^{|\mathcal{L}|} v_\ell^i z_\ell^i$ produces outcomes whereby each member of every placed family can do at least as well as the lowest employment likelihood of the family. We note that both MAX and MIN have equity connotations: while the former attempts to maximize the number of families in which at least one refugee is likely to gain employment, the latter seeks to maximize the number of families in which all of the adults have some chance of getting employment. Finally, defining $v_\ell^i = 1/|N_w^i| \sum_{j \in N_w^i} q_\ell^{ij}$ (MEAN) assigns to v_ℓ^i the average employment likelihood of the (working-age) members in family F^i for affiliate ℓ. Maximizing $\sum_{i=1}^{|\mathcal{F}|} \sum_{\ell=1}^{|\mathcal{L}|} v_\ell^i z_\ell^i$ in this context can be interpreted as placing families into communities so as to ensure that the sum of mean likelihoods is as high as possible.

The outcomes from optimizing using the MAX, MIN, and MEAN objectives are presented in Tables D.1, D.2, and D.3, which are organized in the same manner as Table 1.2. Using the actual FY17 data, the count of cases for which at least one refugee was employed within 90 days was 151, whereas the baseline employment levels when evaluating the actual (manual) placements of cases to affiliates using these alternative objectives are 137.61 for MAX, 97.57 for MIN, and 117.49 for MEAN.

Overall, they exhibit similar performance to the SUM objective that is detailed in Table 1.2. With the exception of one test scenario for the MIN objective (at 62.74 seconds), all combined

Capacity Adjustment	Min Avg Case Size	Binary Service Constraints	Total Expected Employed Refugees	Gains wrt to Predicted Employed Refugees (137.61)	St Dev in Avg Case Size Across Affiliates	# of Unplaced Cases / Refugees	#of Affiliates Violating 90% Capacity	# and % of Cases/Refugees Violating Constraints	Build and Run Time (s)
Observed	None	Off	184.68	34.21%	1.33	0/0	0	55/160 (16.72%/19.07%)	0.48
Observed	None	On	180.98	31.52%	1.28	3/10	0	0/0 (0.00%/0.00%)	0.36
Observed	2	Off	173.98	26.43%	0.81	1/1	0	76/205 (23.10%/24.43%)	0.92
Observed	2	On	171.22	24.43%	1.16	3/10	1	0/0 (0.00%/0.00%)	0.93
Observed	2.5	Off	163.18	18.58%	0.36	4/4	0	90/222 (27.36%/26.46%)	5.43
Observed	2.5	On	160.98	16.98%	0.20	3/7	0	0/0 (0.00%/0.00%)	3.87
Observed	3	Off	138.74	0.82%	0.65	79/89	8	54/153 (16.41%/18.24%)	7.78
Observed	3	On	137.27	-0.25%	0.00	80/92	6	0/0 (0.00%/0.00%)	4.97
Observed	Observed	Off	166.41	20.93%	0.84	2/2	0	83/222 (25.23%/26.46%)	5.40
Observed	Observed	On	164.30	19.39%	1.08	5/6	1	0/0 (0.00%/0.00%)	2.80
$\leq 110\%$	None	Off	188.36	36.88%	1.29	0/0	2	68/214 (20.67%/25.51%)	0.78
$\leq 110\%$	None	On	184.55	34.11%	1.31	2/9	3	0/0 (0.00%/0.00%)	0.55
$\leq 110\%$	2	Off	179.45	30.41%	1.11	0/0	3	70/188 (21.28%/22.41%)	1.09
$\leq 110\%$	2	On	176.23	28.06%	1.11	2/9	4	0/0 (0.00%/0.00%)	0.85
$\leq 110\%$	2.5	Off	168.34	22.33%	0.34	0/0	3	85/199 (25.84%/23.72%)	5.19
$\leq 110\%$	2.5	On	166.02	20.64%	0.59	2/3	6	0/0 (0.00%/0.00%)	3.23
$\leq 110\%$	3	Off	142.64	3.66%	0.00	79/89	5	66/205 (20.06%/24.43%)	5.66
$\leq 110\%$	3	On	140.99	2.45%	0.65	80/92	5	0/0 (0.00%/0.00%)	3.58
$\leq 110\%$	Observed	Off	170.47	23.88%	1.05	0/0	4	83/210 (25.23%/25.03%)	6.35
$\leq 110\%$	Observed	On	168.28	22.28%	1.09	2/3	4	0/0 (0.00%/0.00%)	8.03
$[90\%, 110\%]$	None	Off	188.36	36.88%	1.32	0/0	0	60/176 (18.24%/20.98%)	1.23
$[90\%, 110\%]$	None	On	184.49	34.07%	1.28	1/2	0	0/0 (0.00%/0.00%)	1.01
$[90\%, 110\%]$	2	Off	179.45	30.41%	0.97	0/0	0	73/183 (22.19%/21.81%)	1.40
$[90\%, 110\%]$	2	On	176.14	28.00%	1.05	1/2	0	0/0 (0.00%/0.00%)	1.67
$[90\%, 110\%]$	2.5	Off	168.34	22.33%	0.34	0/0	0	78/198 (23.71%/23.60%)	6.57
$[90\%, 110\%]$	2.5	On	166.02	20.64%	0.33	2/3	0	0/0 (0.00%/0.00%)	5.95
$[90\%, 110\%]$	3	Off			Infeasible instance				
$[90\%, 110\%]$	3	On			Infeasible instance				
$[90\%, 110\%]$	Observed	Off	170.47	23.88%	0.89	5/5	0	73/204 (22.19%/24.31%)	24.71
$[90\%, 110\%]$	Observed	On	168.24	22.26%	1.10	6/7	0	0/0 (0.00%/0.00%)	12.60

Table D.1: Results of counterfactual employment optimization under various scenarios using the MAX objective and LASSO model.

Capacity Adjustment	Min Avg Case Size	Binary Service Constraints	Total Expected Employed Refugees	Gains wrt to Predicted Employed Refugees (97.57)	St Dev in Avg Case Size Across Affiliates	# of Unplaced Cases / Refugees	#of Affiliates Violating 90% Capacity	# and % of Cases/Refugees Violating Constraints	Build and Run Time (s)
Observed	None	Off	120.60	23.60%	1.23	0/0	0	58/160 (17.63%/19.07%)	0.46
Observed	None	On	118.17	21.12%	1.22	1/2	0	0/0 (0.00%/0.00%)	0.43
Observed	2	Off	109.42	12.15%	0.86	1/5	0	75/210 (22.80%/25.03%)	10.75
Observed	2	On	107.38	10.06%	0.78	4/11	1	0/0 (0.00%/0.00%)	3.27
Observed	2.5	Off	98.27	0.72%	0.33	1/5	0	84/242 (25.53%/28.84%)	30.56
Observed	2.5	On	97.12	-0.46%	0.19	7/19	1	0/0 (0.00%/0.00%)	11.66
Observed	3	Off	76.14	-21.96%	0.00	88/116	10	55/176 (16.72%/20.98%)	14.07
Observed	3	On	75.38	-22.75%	0.65	88/116	7	0/0 (0.00%/0.00%)	9.00
Observed	Observed	Off	101.60	4.13%	0.85	2/3	0	83/238 (25.23%/28.37%)	19.68
Observed	Observed	On	100.45	2.95%	0.84	7/19	1	0/0 (0.00%/0.00%)	8.46
$\leq 110\%$	None	Off	120.99	24.00%	1.32	0/0	1	66/189 (20.06%/22.53%)	0.33
$\leq 110\%$	None	On	118.57	21.52%	1.22	1/2	3	0/0 (0.00%/0.00%)	0.34
$\leq 110\%$	2	Off	111.57	14.35%	0.82	0/0	1	71/206 (21.58%/24.55%)	5.75
$\leq 110\%$	2	On	109.52	12.25%	0.86	1/2	4	0/0 (0.00%/0.00%)	1.09
$\leq 110\%$	2.5	Off	101.15	3.67%	0.32	0/0	1	83/223 (25.23%/26.58%)	14.18
$\leq 110\%$	2.5	On	99.84	2.33%	0.29	6/15	3	0/0 (0.00%/0.00%)	6.83
$\leq 110\%$	3	Off	77.61	-20.46%	0.00	89/119	4	58/178 (17.63%/21.22%)	6.95
$\leq 110\%$	3	On	76.98	-21.11%	0.65	89/119	5	0/0 (0.00%/0.00%)	5.45
$\leq 110\%$	Observed	Off	103.67	6.26%	0.90	0/0	2	78/223 (23.71%/26.58%)	19.99
$\leq 110\%$	Observed	On	102.52	5.07%	0.91	7/19	3	0/0 (0.00%/0.00%)	8.32
$[90\%, 110\%]$	None	Off	120.99	24.00%	1.15	0/0	0	68/195 (20.67%/23.24%)	0.44
$[90\%, 110\%]$	None	On	118.57	21.52%	1.16	1/2	0	0/0 (0.00%/0.00%)	0.54
$[90\%, 110\%]$	2	Off	111.57	14.35%	0.79	0/0	0	73/215 (22.19%/25.63%)	8.31
$[90\%, 110\%]$	2	On	109.52	12.25%	0.89	1/2	0	0/0 (0.00%/0.00%)	1.36
$[90\%, 110\%]$	2.5	Off	101.15	3.67%	0.32	0/0	0	93/247 (28.27%/29.44%)	14.36
$[90\%, 110\%]$	2.5	On	99.70	2.18%	0.33	3/5	0	0/0 (0.00%/0.00%)	40.22
$[90\%, 110\%]$	3	Off			Infeasible instance				
$[90\%, 110\%]$	3	On			Infeasible instance				
$[90\%, 110\%]$	Observed	Off	103.67	6.26%	0.83	1/1	0	84/229 (25.53%/27.29%)	62.74
$[90\%, 110\%]$	Observed	On	102.44	5.00%	0.86	5/9	0	0/0 (0.00%/0.00%)	37.80

Table D.2: Results of counterfactual employment optimization under various scenarios using the MIN objective and LASSO model.

build and solve runtimes complete in well under one minute. We observe that the same test scenarios seem to perform comparatively well across all of the objectives. That said, for the MEAN objective some test scenarios approach and even exceed 40% gains. Unsurprisingly, the gains for the MIN objective are comparatively lower, and for a few of the scenarios with very restrictive constraints (e.g. minimum average case size of 3), negative gains are apparent in some of the objectives. Importantly, these low gain scenarios are precisely the instances for

Capacity Adjustment	Min Avg Case Size	Binary Service Constraints	Total Expected Employed Refugees	Gains wrt to Predicted Employed Refugees (117.49)	St Dev in Avg Case Size Across Affiliates	# of Unplaced Cases / Refugees	#of Affiliates Violating 90% Capacity	# and % of Cases/Refugees Violating Constraints	Build and Run Time (s)
Observed	None	Off	163.81	39.42%	1.39	0/0	0	65/200 (19.76%/23.84%)	1.62
Observed	None	On	160.67	36.75%	1.30	2/9	0	0/0 (0.00%/0.00%)	0.49
Observed	2	Off	151.73	29.14%	0.79	3/3	0	77/219 (23.40%/26.10%)	2.43
Observed	2	On	149.16	26.96%	1.06	3/10	1	0/0 (0.00%/0.00%)	1.46
Observed	2.5	Off	140.19	19.32%	0.45	4/4	0	91/245 (27.66%/29.20%)	12.60
Observed	2.5	On	138.07	17.51%	0.46	5/6	0	0/0 (0.00%/0.00%)	8.57
Observed	3	Off	115.36	-1.81%	0.00	80/92	7	60/186 (18.24%/22.17%)	5.92
Observed	3	On	113.87	-3.08%	0.65	81/95	5	0/0 (0.00%/0.00%)	3.66
Observed	Observed	Off	143.51	22.15%	0.85	2/2	0	79/220 (24.01%/26.22%)	7.52
Observed	Observed	On	141.50	20.44%	0.84	4/10	2	0/0 (0.00%/0.00%)	4.46
$\leq 110\%$	None	Off	166.39	41.62%	1.40	0/0	1	63/191 (19.15%/22.77%)	0.66
$\leq 110\%$	None	On	163.17	38.88%	1.47	2/9	2	0/0 (0.00%/0.00%)	1.16
$\leq 110\%$	2	Off	156.33	33.06%	1.48	0/0	4	72/210 (21.88%/25.03%)	1.42
$\leq 110\%$	2	On	153.41	30.57%	1.40	2/9	2	0/0 (0.00%/0.00%)	1.38
$\leq 110\%$	2.5	Off	144.58	23.06%	0.32	0/0	1	88/228 (26.75%/27.18%)	13.24
$\leq 110\%$	2.5	On	142.31	21.12%	0.34	2/3	2	0/0 (0.00%/0.00%)	5.34
$\leq 110\%$	3	Off	118.41	0.79%	0.00	80/92	4	64/193 (19.45%/23.00%)	7.34
$\leq 110\%$	3	On	116.85	-0.55%	0.90	81/95	9	0/0 (0.00%/0.00%)	6.30
$\leq 110\%$	Observed	Off	146.87	25.00%	0.90	0/0	4	69/198 (20.97%/23.60%)	10.13
$\leq 110\%$	Observed	On	144.83	23.27%	0.99	2/3	5	0/0 (0.00%/0.00%)	7.52
$[90\%, 110\%]$	None	Off	166.39	41.62%	1.38	0/0	0	57/162 (17.33%/19.31%)	1.32
$[90\%, 110\%]$	None	On	163.16	38.87%	1.26	1/2	0	0/0 (0.00%/0.00%)	1.42
$[90\%, 110\%]$	2	Off	156.33	33.06%	0.94	0/0	0	79/215 (24.01%/25.63%)	1.53
$[90\%, 110\%]$	2	On	153.39	30.56%	0.99	1/2	0	0/0 (0.00%/0.00%)	1.76
$[90\%, 110\%]$	2.5	Off	144.58	23.06%	0.34	0/0	0	88/242 (26.75%/28.84%)	14.74
$[90\%, 110\%]$	2.5	On	142.31	21.12%	0.33	2/3	0	0/0 (0.00%/0.00%)	5.20
$[90\%, 110\%]$	3	Off			Infeasible instance				
$[90\%, 110\%]$	3	On			Infeasible instance				
$[90\%, 110\%]$	Observed	Off	146.87	25.00%	0.83	1/1	0	88/240 (26.75%/28.61%)	8.88
$[90\%, 110\%]$	Observed	On	144.76	23.21%	0.86	2/3	0	0/0 (0.00%/0.00%)	12.13

Table D.3: Results of counterfactual employment optimization under various scenarios using the MEAN objective and LASSO model.

which many cases and refugees remained unplaced, thus excluding their contribution in the objective.

			Relative to SUM Objective				Relative to MAX Objective				Relative to MIN Objective				Relative to MEAN Objective			
Capacity Adjustment	Min Avg Case Size	Binary Service Constraints	SUM	MAX (% loss)	MIN (% loss)	MEAN (% loss)	MAX	SUM (% loss)	MIN (% loss)	MEAN (% loss)	MIN	SUM (% loss)	MAX (% loss)	MEAN (% loss)	MEAN	SUM (% loss)	MAX (% loss)	MIN (% loss)
Observed	None	Off	**213.02**	207.64 (2.53%)	193.82 (9.02%)	209.80 (1.51%)	**184.68**	180.63 (2.19%)	169.90 (8.01%)	183.10 (0.86%)	**120.60**	116.14 (3.70%)	117.85 (2.28%)	118.91 (1.40%)	**163.81**	161.00 (1.71%)	162.47 (0.82%)	152.73 (6.76%)
Observed	None	On	**208.25**	203.85 (2.11%)	192.89 (7.37%)	205.40 (1.37%)	**180.98**	177.16 (2.11%)	168.67 (6.80%)	179.03 (1.08%)	**118.17**	114.53 (3.08%)	115.40 (2.35%)	116.63 (1.30%)	**160.67**	157.93 (1.70%)	159.18 (0.92%)	151.23 (5.88%)
Observed	2	Off	**206.28**	201.65 (2.24%)	181.75 (11.89%)	201.73 (2.20%)	**173.98**	170.38 (2.07%)	158.94 (8.64%)	172.04 (1.11%)	**109.42**	102.17 (6.63%)	101.14 (7.57%)	102.86 (6.00%)	**151.73**	149.70 (1.34%)	150.18 (1.02%)	141.15 (6.97%)
Observed	2	On	**202.03**	197.48 (2.25%)	181.13 (10.34%)	197.70 (2.14%)	**171.22**	168.16 (1.79%)	158.25 (7.58%)	169.35 (1.10%)	**107.38**	100.48 (6.43%)	99.93 (6.94%)	100.98 (5.96%)	**149.16**	147.60 (1.05%)	147.79 (0.92%)	139.62 (6.40%)
Observed	2.5	Off	**196.76**	191.02 (2.92%)	173.53 (11.81%)	192.09 (2.37%)	**163.18**	159.43 (2.30%)	150.87 (7.54%)	161.25 (1.18%)	**98.27**	88.12 (10.32%)	88.63 (9.81%)	90.22 (8.19%)	**140.19**	137.47 (1.94%)	138.05 (1.53%)	131.79 (5.99%)
Observed	2.5	On	**192.95**	187.22 (2.97%)	170.86 (11.45%)	187.03 (3.07%)	**160.98**	157.47 (2.18%)	147.91 (8.12%)	158.99 (1.24%)	**97.12**	87.53 (9.87%)	87.94 (9.45%)	89.89 (7.44%)	**138.07**	135.70 (1.71%)	136.14 (1.39%)	129.42 (6.27%)
Observed	3	Off	**172.83**	167.26 (3.22%)	147.03 (14.93%)	167.45 (3.11%)	**138.74**	134.78 (2.85%)	124.41 (10.33%)	136.52 (1.60%)	**76.14**	63.52 (16.57%)	63.28 (16.90%)	66.02 (13.29%)	**115.36**	112.37 (2.59%)	113.10 (1.96%)	106.07 (8.05%)
Observed	3	On	**169.64**	163.72 (3.49%)	146.33 (13.74%)	164.45 (3.06%)	**137.27**	134.28 (2.18%)	124.00 (9.66%)	134.58 (1.96%)	**75.38**	63.19 (16.17%)	63.67 (15.54%)	65.53 (13.07%)	**113.87**	111.49 (2.09%)	111.85 (1.78%)	105.28 (7.55%)
Observed	Observed	Off	**199.33**	193.90 (2.73%)	177.11 (11.15%)	193.66 (2.85%)	**166.41**	162.79 (2.18%)	153.71 (7.63%)	164.50 (1.15%)	**101.60**	92.63 (8.82%)	92.41 (9.05%)	94.47 (7.02%)	**143.51**	141.11 (1.67%)	141.63 (1.31%)	134.62 (6.20%)
Observed	Observed	On	**195.65**	190.42 (2.67%)	175.11 (10.50%)	190.19 (2.79%)	**164.30**	161.05 (1.98%)	152.96 (6.90%)	162.38 (1.17%)	**100.45**	92.03 (8.38%)	91.64 (8.77%)	93.09 (7.33%)	**141.50**	139.48 (1.43%)	139.72 (1.26%)	133.56 (5.61%)
$\leq 110\%$	None	Off	**218.05**	212.38 (2.60%)	197.89 (9.25%)	215.37 (1.23%)	**188.36**	184.24 (2.18%)	172.78 (8.27%)	186.45 (1.01%)	**120.99**	117.57 (2.83%)	118.21 (2.30%)	119.21 (1.47%)	**166.39**	163.98 (1.45%)	165.04 (0.81%)	154.64 (7.06%)
$\leq 110\%$	None	On	**212.96**	208.12 (2.27%)	195.13 (8.37%)	210.51 (1.15%)	**184.55**	181.21 (1.81%)	170.71 (7.50%)	182.71 (1.00%)	**118.57**	115.34 (2.72%)	115.98 (2.18%)	117.02 (1.31%)	**163.17**	160.95 (1.36%)	161.73 (0.88%)	152.80 (6.35%)
$\leq 110\%$	2	Off	**212.39**	207.34 (2.38%)	185.02 (12.89%)	208.25 (1.95%)	**179.45**	175.71 (2.09%)	161.95 (9.76%)	177.61 (1.03%)	**111.57**	105.13 (5.77%)	103.96 (6.82%)	105.73 (5.23%)	**156.33**	154.47 (1.19%)	154.53 (1.15%)	143.69 (8.08%)
$\leq 110\%$	2	On	**207.72**	202.57 (2.48%)	184.66 (11.10%)	203.97 (1.80%)	**176.23**	173.18 (1.73%)	162.19 (7.96%)	174.35 (1.07%)	**109.52**	103.46 (5.53%)	101.99 (6.87%)	103.55 (5.45%)	**153.41**	151.82 (1.03%)	151.84 (1.02%)	142.83 (6.90%)
$\leq 110\%$	2.5	Off	**202.75**	196.42 (3.12%)	176.22 (13.08%)	198.17 (2.26%)	**168.34**	164.77 (2.12%)	153.79 (8.64%)	166.24 (1.25%)	**101.15**	91.11 (9.93%)	90.97 (10.07%)	93.67 (7.40%)	**144.58**	142.02 (1.77%)	142.28 (1.59%)	134.75 (6.80%)
$\leq 110\%$	2.5	On	**198.84**	192.86 (3.00%)	176.14 (11.42%)	194.09 (2.39%)	**166.02**	162.92 (1.87%)	152.72 (8.01%)	163.82 (1.32%)	**99.84**	90.21 (9.65%)	90.14 (9.71%)	92.37 (7.49%)	**142.30**	140.22 (1.47%)	140.41 (1.33%)	133.57 (6.14%)
$\leq 110\%$	3	Off	**177.51**	172.20 (2.99%)	149.22 (15.94%)	172.23 (2.97%)	**142.64**	139.04 (2.52%)	126.76 (11.13%)	140.06 (1.81%)	**77.60**	64.83 (16.46%)	64.74 (16.58%)	67.43 (13.11%)	**118.41**	115.34 (2.59%)	116.19 (1.88%)	108.34 (8.51%)
$\leq 110\%$	3	On	**174.27**	168.96 (3.05%)	147.58 (15.32%)	169.80 (2.56%)	**140.99**	137.76 (2.29%)	126.53 (10.25%)	138.35 (1.87%)	**76.98**	64.89 (15.70%)	65.06 (15.48%)	67.85 (11.86%)	**116.85**	114.24 (2.23%)	114.88 (1.68%)	107.65 (7.87%)
$\leq 110\%$	Observed	Off	**204.27**	198.15 (2.99%)	181.01 (11.39%)	198.62 (2.77%)	**170.47**	166.62 (2.26%)	157.26 (7.75%)	168.20 (1.33%)	**103.67**	94.51 (8.84%)	93.92 (9.41%)	96.70 (6.73%)	**146.87**	144.47 (1.63%)	144.66 (1.50%)	137.91 (6.10%)
$\leq 110\%$	Observed	On	**200.49**	194.91 (2.79%)	178.71 (10.86%)	196.01 (2.23%)	**168.28**	164.94 (1.98%)	154.97 (7.91%)	166.08 (1.31%)	**102.52**	93.84 (8.46%)	92.99 (9.29%)	95.17 (7.17%)	**144.83**	142.59 (1.55%)	143.07 (1.22%)	136.02 (6.08%)
[90%, 110%]	None	Off	**218.05**	212.38 (2.60%)	194.95 (10.60%)	215.37 (1.23%)	**188.36**	184.24 (2.18%)	171.00 (9.22%)	186.45 (1.01%)	**120.99**	117.57 (2.83%)	118.21 (2.30%)	119.21 (1.47%)	**166.39**	163.98 (1.45%)	165.04 (0.81%)	153.11 (7.98%)
[90%, 110%]	None	On	**212.91**	208.03 (2.29%)	193.31 (9.21%)	210.61 (1.08%)	**184.49**	181.12 (1.82%)	168.79 (8.51%)	182.53 (1.06%)	**118.57**	115.33 (2.73%)	115.98 (2.18%)	116.89 (1.41%)	**163.16**	160.92 (1.37%)	161.69 (0.90%)	151.82 (6.95%)
[90%, 110%]	2	Off	**212.39**	207.34 (2.38%)	187.11 (11.90%)	208.25 (1.95%)	**179.45**	175.71 (2.09%)	163.60 (8.83%)	177.61 (1.03%)	**111.57**	105.13 (5.77%)	103.96 (6.82%)	105.73 (5.23%)	**156.33**	154.47 (1.19%)	154.53 (1.15%)	144.63 (7.49%)
[90%, 110%]	2	On	**207.58**	202.61 (2.39%)	184.64 (11.05%)	203.90 (1.77%)	**176.14**	173.08 (1.74%)	161.48 (8.32%)	174.25 (1.07%)	**109.52**	103.66 (5.35%)	102.31 (6.58%)	103.60 (5.41%)	**153.39**	151.92 (0.96%)	152.12 (0.83%)	142.64 (7.01%)
[90%, 110%]	2.5	Off	**202.75**	196.42 (3.12%)	178.99 (11.72%)	198.17 (2.26%)	**168.34**	164.77 (2.12%)	155.58 (7.58%)	166.24 (1.25%)	**101.15**	91.11 (9.93%)	90.97 (10.07%)	93.67 (7.40%)	**144.58**	142.02 (1.77%)	142.28 (1.59%)	135.95 (5.97%)
[90%, 110%]	2.5	On	**198.81**	192.86 (2.99%)	176.69 (11.13%)	194.09 (2.38%)	**166.02**	162.75 (1.97%)	152.98 (7.85%)	163.82 (1.32%)	**99.70**	90.40 (9.32%)	90.14 (9.58%)	92.37 (7.35%)	**142.30**	140.33 (1.39%)	140.41 (1.33%)	133.65 (6.08%)
[90%, 110%]	3	Off	Infeasible instance															
[90%, 110%]	3	On	Infeasible instance															
[90%, 110%]	Observed	Off	**204.26**	198.15 (2.99%)	179.80 (11.98%)	198.62 (2.76%)	**170.47**	166.80 (2.16%)	156.74 (8.06%)	168.20 (1.33%)	**103.67**	94.77 (8.59%)	93.92 (9.41%)	96.70 (6.73%)	**146.87**	144.69 (1.48%)	144.66 (1.50%)	137.34 (6.49%)
[90%, 110%]	Observed	On	**200.36**	194.91 (2.72%)	179.09 (10.62%)	196.31 (2.02%)	**168.24**	165.07 (1.89%)	156.38 (7.05%)	166.10 (1.27%)	**102.44**	94.16 (8.09%)	92.99 (9.23%)	95.06 (7.21%)	**144.76**	142.75 (1.39%)	143.01 (1.21%)	136.39 (5.78%)

Table D.4: Comparison of performance of alternative SUM, MAX, MIN, and MEAN objectives of interest. Analysis is with respect to the upper headings and four subcolumns. Under each of these headings, entries in **bold** are the maximized value according to that objective, and remaining columns report the performance of the optimal solution of each alternative objective, when evaluated according to the objective of the upper heading, along with associated percent loss.

Table D.4 reports how the optimal solutions obtained by optimizing using the SUM, MAX, MIN, or MEAN objectives perform with respect to being evaluated in the alternative objectives, for each of the 30 test scenarios. In particular, we organize Table D.4 so that for each of these objectives, one can readily compare the objective function values for the various optimal solutions. We also report the percent loss in objective function value from using an optimal solution to an alternative objective function.

As can be observed in Table D.4, in general the optimal solutions to alternative objective functions perform quite well across all experiments—typically experiencing a loss of only a few percent when comparing SUM, MAX, and MEAN. The exception is the MIN objective, where more variation is observed—either in the high single digits or low double digits, up to a max of 16.9% under the test scenario with observed capacity, minimum average case size of three, de-activated binary service constraints, and evaluating the optimal solution to the MAX objective in the MIN objective. With respect to the MIN objective, unsurprisingly the evaluation in the MIN objective of optimal solutions obtained from optimizing the MEAN objective seem to perform best, with all percent losses in the single digits.

Appendix E

Chapter 1: Counterfactual Optimization Experiments with Uncertainty around Predictions

In this section we allow for uncertainty in the modeling environment as it pertains to estimating the quality score for each refugee and affiliate. We evaluate the performance of our optimized placement outcome $z^\star$ with respect to alternative objective functions sampled from the same distribution of estimated probabilities. In other words, we hold the optimal refugee allocation fixed, and evaluate the change in employment gains as we perturb and allow for errors in the estimated employment probabilities.

The bootstrapping approach described in Section 1.4 generated 1,000 sets of models that predict alternative employment probabilities for each refugee-affiliate pair. This exercise generates 1,000 objective functions, all of which have slightly different quality scores than that of the objective function obtained from the original estimated $\hat{q}_\ell^{ij}$ quality scores. To assess the performance of our optimization with respect to the uncertainty in predicted employment likelihoods, we then evaluate the optimized placement outcome $z^\star$ using each of the 1,000 bootstrapped instances for each of the thirty test scenarios.

We obtain a distribution of employment gains, which we would expect to produce lower employment gains on average than if our employment probabilities were predicted exactly, as in this latter case the optimal allocation is obtained using optimization. Nonetheless, this exercise shows that our approach produces remarkably stable employment gains even under uncertainty.

Figure E.1 depicts box plots for 10 of the 30 scenarios for which there were feasible solutions. These box plots show the distribution of employment gains given uncertainty in our predicted employment probabilities, holding the optimal allocation fixed. The figure shows that the performance of $z^\star$ clusters tightly around the median values, and is well within the first and third quartiles. The performance of $z^\star$ exceeds the median since it was obtained using optimization. However, the average employment gains under uncertainty are very near (within 2% for most scenarios) to those obtained assuming that the predicted probabilities were at their means. We can therefore conclude that optimal allocations computed by our approach produce stable employment gains.

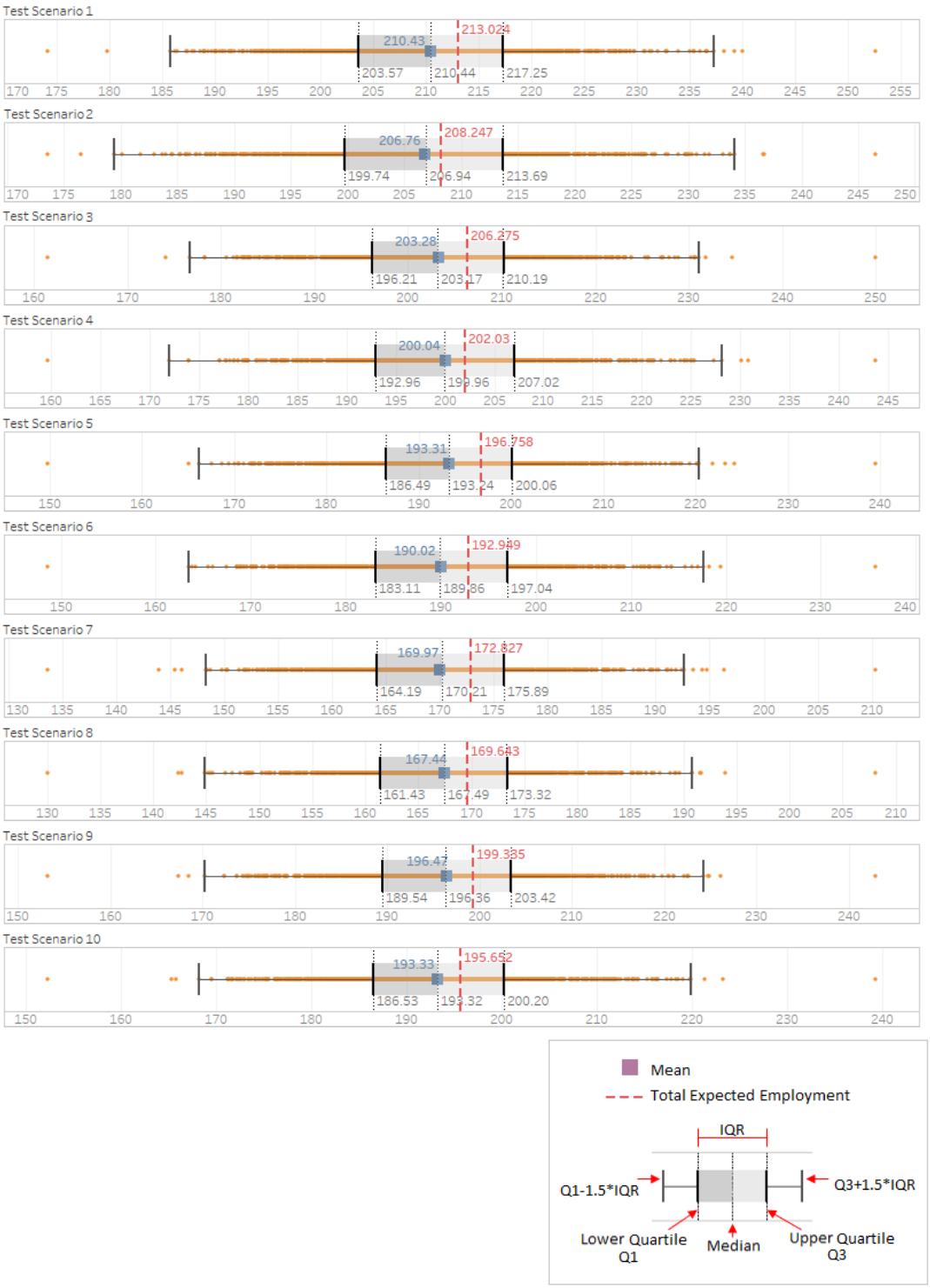


Figure E.1: Uncertainty in the objective function illustrated across the first ten scenarios, namely those that use "Observed" capacity levels. Box plots depicting the distribution of the evaluation of the optimized $z^\star$ solution in 1,000 bootstrapped objective functions.

Appendix F

Chapter 1: Counterfactual Optimization Experiments with Multiple Placement Periods

To evaluate the performance of our approach with respect to inherent uncertainty in refugee arrivals, we consider counterfactual optimization of the SUM objective with $n > 1$ placement periods in FY17 data. Specifically, we consider $n = \{4, 12, 52\}$ placement periods over the 30 test scenarios, and compare with $n = 1$, that is, the results detailed in Table 1.2.

Given the limited size of the data, such experiments introduced some nuanced challenges that required creative handling. In particular, larger values of n may cause insufficient per-period capacities at certain affiliates to accommodate certain families having many members. This situation is further compounded by single-person families that have relatively high employment likelihoods, that are simply able to fill (pack) the available capacity at affiliates that offer superior employment prospects. Note that this was rarely the case under manual placements: affiliate capacity was regularly exceeded for any given period, so long as later periods use respectively less capacity.

Therefore, in the following experiments we incorporated some guidelines to accommodate the above challenges. All refugee families that are unplaced in a given period are included in the next placement period. For each period, the total period capacity over all affiliates is set to the number of arriving refugees for that placement period, in addition to any unplaced refugees from earlier periods. The per-period capacity for each affiliate is then obtained by simply scaling the total period capacity by the affiliate's annual observed proportion (and rounding to the nearest integer). For any placement period, if any affiliate has already reached its annual capacity, its capacity for the present and future placement periods is set to zero, and the proportional share of per-period capacity is evenly distributed among other affiliates that have not yet used their annual capacity.

			Yearly ($n = 1$, Single Optimization)			Quarterly ($n = 4$ Optimizations)			Monthly ($n = 12$ Optimizations)			Weekly ($n = 52$ Optimizations)		
Capacity Adjustment	Min Avg Case Size	Binary Service Constraints	Total Expected Employed Refugees	Gains wrt to Predicted Employed Refugees (157.93)	# of Unplaced Cases / Refugees	Total Expected Employed Refugees	Gains wrt to Predicted Employed Refugees (157.93)	# of Unplaced Cases / Refugees	Total Expected Employed Refugees	Gains wrt to Predicted Employed Refugees (157.93)	# of Unplaced Cases / Refugees	Total Expected Employed Refugees	Gains wrt to Predicted Employed Refugees (157.93)	# of Unplaced Cases / Refugees
Observed	None	Off	213.024	34.89%	0/0	209.875	32.89%	4/23	207.985	31.69%	3/18	194.658	23.26%	3/19
Observed	None	On	208.247	31.86%	3/10	206.127	30.52%	7/29	201.633	27.67%	8/38	183.349	16.10%	21/59
Observed	2	Off	206.275	30.61%	1/1	198.688	25.81%	7/27	193.738	22.67%	10/10	182.168	15.35%	14/14
Observed	2	On	202.03	27.92%	2/9	195.252	23.63%	11/38	187.07	18.45%	18/44	169.204	7.14%	36/68
Observed	2.5	Off	196.758	24.59%	1/1	186.658	18.19%	24/37	179.93	13.93%	35/37	164.228	3.99%	58/58
Observed	2.5	On	192.949	22.17%	3/7	182.574	15.60%	28/43	171.824	8.80%	42/58	153.915	-2.54%	73/103
Observed	3	Off	172.827	9.43%	78/86	167.598	6.12%	78/86	161.751	2.42%	78/82	146.847	-7.02%	97/101
Observed	3	On	169.643	7.42%	79/89	165.656	4.89%	78/85	153.858	-2.58%	90/107	136.769	-13.40%	109/139
Observed	Observed	Off	199.335	26.22%	2/2	189.913	20.25%	24/27	178.564	13.07%	41/43	164.037	3.87%	62/67
Observed	Observed	On	195.652	23.89%	4/8	183.585	16.24%	31/51	171.916	8.86%	49/67	156.85	-0.68%	73/107
$\leq 110\%$	None	Off	218.055	38.07%	0/0	215.816	36.65%	2/12	210.658	33.39%	4/24	198.996	26.00%	6/37
$\leq 110\%$	None	On	212.959	34.84%	2/9	210.652	33.38%	4/18	205.737	30.27%	7/38	190.165	20.41%	17/43
$\leq 110\%$	2	Off	212.389	34.48%	0/0	206.649	30.85%	4/14	193.656	22.62%	11/39	181.962	15.22%	30/50
$\leq 110\%$	2	On	207.72	31.53%	2/9	203.967	29.15%	4/22	192.196	21.70%	12/40	177.023	12.09%	36/56
$\leq 110\%$	2.5	Off	202.75	28.38%	0/0	194.902	23.41%	14/20	178.237	12.86%	38/49	168.257	6.54%	56/57
$\leq 110\%$	2.5	On	198.837	25.90%	3/7	189.295	19.86%	20/33	177.391	12.32%	36/49	160.813	1.83%	65/89
$\leq 110\%$	3	Off	177.511	12.40%	78/86	172.399	9.16%	78/86	166.175	5.22%	78/84	151.179	-4.28%	93/93
$\leq 110\%$	3	On	174.266	10.34%	79/89	169.213	7.15%	81/91	161.88	2.50%	84/93	142.874	-9.53%	107/136
$\leq 110\%$	Observed	Off	204.268	29.34%	0/0	196.716	24.56%	16/22	180.308	14.17%	40/45	168.245	6.53%	55/56
$\leq 110\%$	Observed	On	200.49	26.95%	3/7	191.986	21.56%	19/37	176.133	11.53%	42/61	163.035	3.23%	67/95
$[90\%, 110\%]$	None	Off	218.055	38.07%	0/0	Infeasible instance			Infeasible instance			Infeasible instance		
$[90\%, 110\%]$	None	On	212.914	34.82%	1/2	Infeasible instance			Infeasible instance			Infeasible instance		
$[90\%, 110\%]$	2	Off	212.389	34.48%	0/0	Infeasible instance			Infeasible instance			Infeasible instance		
$[90\%, 110\%]$	2	On	207.575	31.44%	2/6	Infeasible instance			Infeasible instance			Infeasible instance		
$[90\%, 110\%]$	2.5	Off	202.75	28.38%	0/0	Infeasible instance			Infeasible instance			Infeasible instance		
$[90\%, 110\%]$	2.5	On	198.81	25.89%	2/3	Infeasible instance			Infeasible instance			Infeasible instance		
$[90\%, 110\%]$	3	Off	Infeasible instance			Infeasible instance			Infeasible instance			Infeasible instance		
$[90\%, 110\%]$	3	On	Infeasible instance			Infeasible instance			Infeasible instance			Infeasible instance		
$[90\%, 110\%]$	Observed	Off	204.259	29.34%	5/5	Infeasible instance			Infeasible instance			Infeasible instance		
$[90\%, 110\%]$	Observed	On	200.357	26.86%	6/7	Infeasible instance			Infeasible instance			Infeasible instance		

Table F.1: Results of counterfactual employment optimization over the entire year, over 4 quarters, over 12 months, and over 52 weeks using the SUM objective and LASSO model.

Table F.1 details the results of our experiments and is organized in the same manner as Table 1.2, with each row reporting one of the 30 test scenarios. There are four sets of columns, each of which reports the total expected employed refugees, percent gains over the predicted baseline of 157.93, and number of unplaced cases and refugees. The first set of columns repeats these values first reported in Table 1.2, whereas the last three report these values for $n = 4$, $n = 12$, and $n = 52$, respectively.

As can be seen in Table F.1, the percent gains with respect to predicted employed refugees (157.93) remain high for $n = 4$ and $n = 12$. In particular, for the routine operational scenario that the second row embodies of using observed capacity, no minimum average case size constraints, and activated binary service constraints, we see that the gains drop only slightly from 31.86% for $n = 1$, to 30.52% for $n = 4$, and then again to a respectable 27.67% for $n = 12$. The earlier described effect of insufficient per-period capacities begins to be noticed for $n = 52$, which is why a percent gain of only 16.10% is achieved, and also why there are many more unplaced cases and refugees. In general, greater numbers of unplaced refugees and cases naturally lead to lower gains, which may even be negative in extreme scenarios. With greater numbers of refugees to be placed, we naturally would expect these gains to be higher.

Appendix G

Chapter 2: Data Preprocessing and Employment Prediction

We use arrival data directly collected by HIAS, recent as of Summer 2020. We drop entries in the database missing crucial information: agent and case identifiers, the case's pool (free or with US ties), and the dates of birth, of allocation, and of arrival. We furthermore remove all cases that were allocated before March 2011 since our employment prediction uses local unemployment data, which is incomplete before then. This last step removes 3.5% of refugees from fiscal year 2014, and less than 1% of refugees from the subsequent fiscal years. Due to this removal, we do not include fiscal years before 2014 in our analysis, for which a higher percentage of arrivals would have been removed.

To predict employment scores, we train two separate LASSO models following the methodology of Ahani et al. [61], one model for free cases and one for tied cases. Training separate models is helpful since, as Ahani et al. note, tied cases find employment in different patterns due to existing support networks. The regression is trained on cases allocated in fiscal years 2011 to 2019, where the hyper-parameters are determined via cross validation before retraining on the whole time range. Note that training on years that we evaluate on would be problematic for evaluating how well the employment scores match the ground-truth employment. However, since our evaluations measure how well the matching algorithms do relative to the given employment scores, training on a wide range of years ensures that the employment scores are as accurate as possible.

Appendix H

Chapter 2: Alternative Potential Approach

In section 2.4.1, we presented a procedure for calculating potentials, termed Pot1, which was based on the shadow prices of a matching LP. Here, we present a second such procedure Pot2, which is based on a slightly different LP and has a different theoretical underpinnings:

- whereas the matching LP for Pot1 does not include the current batch, the current batch is included in the LP for Pot2,
- whereas Pot1 uses the element-wise maximal set of shadow prices, Pot2 uses the element-wise minimal one, and
- whereas Pot1 is motivated by two-stage stochastic programming, Pot2 is motivated by a connection to Walrasian equilibria.

The procedure is given in algorithm 4, and it can be immediately plugged into PMB. Note that the dual of the linear program in line 3 looks as follows:

$$\begin{aligned} \text{minimize } & \sum_{i \in t'+1}^{n} y_i + \sum_{\ell \in L} p_\ell & & \\ \text{subject to } & y_i \geq u_{i,\ell} - s_i\, p_\ell & & \forall i = (t'+1), \ldots, n, \forall \ell \in L \\ & p_\ell \geq 0 & & \forall \ell \in L. \end{aligned}$$

Suppose that $s_i = 1$ for all $i \in N$, as we do in section 2.4. Then, for any (non-fractional) optimal matching $x_{i,\ell}$ in the primal, and any set of optimal dual variables p_ℓ, it is well known (and follows from complementary slackness) that the two form a *Walrasian equilibrium*. That is, if we consider $u_{i,\ell}$ as case i's utility for being matched to affiliate ℓ, and if we imagine charging case i a price of p_ℓ for being matched to affiliate ℓ, the optimal matching matches each case i to an affiliate ℓ maximizing the profit $u_{i,\ell} - p_\ell$. Thus, if all trajectories perfectly predicted future arrivals, and if ties were broken in a specific way in each step, equipping PM with the potentials of Pot2 would lead to the optimal-employment matching.

Note that this justification also extends to batching (still assuming $s_i = 1$), since the Walrasian equilibrium shows that it is possible to maximize $u_{i,\ell} - p_i$ for all cases at once. Thus, if the trajectory perfectly anticipates future arrivals, each optimal hindsight matching maximizes the matching ILP allocating each batch in PMB.

When the shadow prices are chosen as the element-wise minimal set of shadow prices, work by Hsu, Morgenstern, Rogers, Roth, and Vohra [96] gives arguments that PM with Pot2 should be somewhat robust to tie breaking, assuming that employment scores satisfy a genericity condition. In addition, they show that, for large matching problems, this matching algorithm should also be robust if trajectories are drawn from the same i.i.d. distribution as the real set of future arrivals.[1]

Like the justification of the Pot2 potentials, this justification does not cleanly extend to cases of non-unit size. Furthermore, there is no immediate theoretical justification for averaging these prices over multiple trajectories, which has to be evaluated empirically.

ALGORITHM 4: Pot2

Parameter: $k \in \mathbb{N}_{\geq 1}$, the number of trajectories per potential computation

Input: remaining capacities $\boldsymbol{c}$, the index $t'+1, \ldots, t$ of the cases in the current batch (see algorithm 3), characteristics of cases arriving in the 6 past months

Output: a set of potentials p_ℓ for all affiliates ℓ

for $j = 1, \ldots, k$ **do**

  for each $i = t+1, \ldots, n$, set s_i and $\{u_{i,\ell}\}_\ell$ to the size and employment scores of a random, recently arrived case;

  solve the following bipartite-matching LP:

$$
\begin{aligned}
\text{maximize} \quad & \sum_{i=t'+1}^{n} \sum_{\ell \in L} u_{i,\ell}\, x_{i,\ell} \\
\text{subject to} \quad & \sum_{\ell \in L} x_{i,\ell} = 1 && \forall i = (t'+1), \ldots, n \\
& \sum_{i=t'+1}^{n} s_i\, x_{i,\ell} \leq c_\ell && \forall \ell \in L \quad (*) \\
& x_{i,\ell} \geq 0 && \forall i = (t'+1), \ldots, n, \forall \ell \in L.
\end{aligned}
$$

  for each ℓ, set p_ℓ^j to the minimal shadow price of the constraint $(*)$ for ℓ;

set $p_\ell \leftarrow (\sum_{j=1}^{k} p_\ell^j)/k$ for all ℓ;

return $\{p_\ell\}_{\ell \in L}$;

[1] This mapping is not perfect since Hsu et al. show that capacities are not exceeded by much, rather than considering cascading effects of refugees switching affiliates once their most preferred affiliate is at capacity. This problem could be remedied by slightly reducing capacities in the computation of shadow prices.

Appendix I

Chapter 2: Additional Experiments

I.0.1 Employment Statistics for Non-Unit Cases without Batching

I.0.2 Timing

In the experiment of section 2.6.1, we measure the running time for a call to PMB, which mainly consists of the time of sampling the k random trajectories, of computing the appropriate shadow prices for each trajectory, and of solving the matching ILP. This time will generally increase the more arrivals are expected in the remainder of the fiscal year since this determines the length of the trajectory and the size of the LP whose duals must be computed. Additionally, batches consisting of more cases should also increase running time since the matching ILP is larger for such batches.

Therefore, we benchmark the algorithms on fiscal year 2016, which saw the largest number of arriving cases (1,628). Within this fiscal year, we aim to pick a batch that is both early and large, which is a bit complicated by the fact that the earliest batches of the year are small. We choose a batch of 31 cases, which comes early enough that 1,486 of the 1,628 cases of the fiscal year come after it. This batch is the 17th largest out of the 168 batches of the year, and all preceding batches are less than half as many cases.

For this batch, we measure the running time for PMB with potentials Pot1 and Pot2, each with $k \in \{1, 5, 9\}$. Each reported running time is the average of 50 iterations and is measured on a 2017 MacBook Pro with a 3.1 GHz Dual-Core Intel Core i5 processor using Gurobi [40] for solving LPs and ILPs:

Matching algorithm	Running time
PMB(Pot1($k = 1$))	0.5 s
PMB(Pot1($k = 5$))	2.4 s
PMB(Pot1($k = 9$))	4.0 s
PMB(Pot2($k = 1$))	0.5 s
PMB(Pot2($k = 5$))	2.6 s
PMB(Pot2($k = 9$))	4.6 s

Given that a resettlement agency needs to execute the matching algorithm at most once per day, these times are negligible. For low k and an optimized implementation, it might even be possible to support near real-time experimentation; for example, an *Annie*™ user could drag

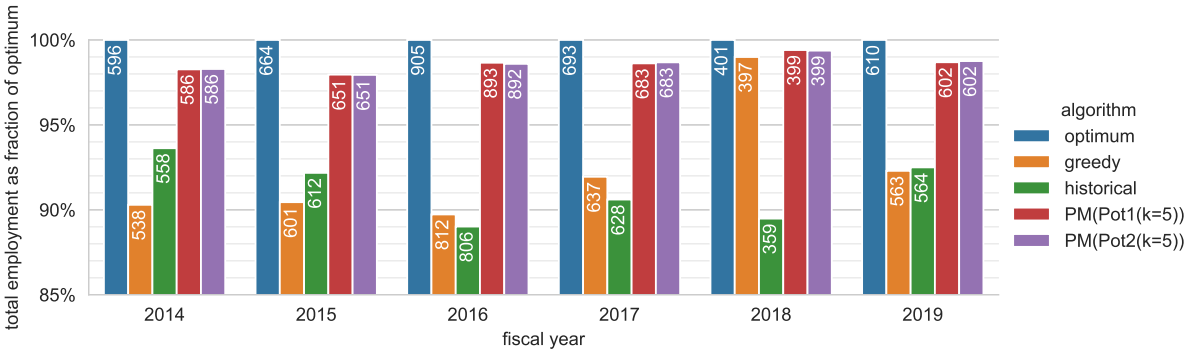


Figure I.1: Total employment obtained by different algorithms, with whole cases arriving rather than being split up. Capacities are the final capacities of the fiscal year. For the potential algorithms, employment is averaged over 10 random runs. The numbers in the bars denote the absolute total employment; the bar height indicates the proportion of the optimum total employment in hindsight.

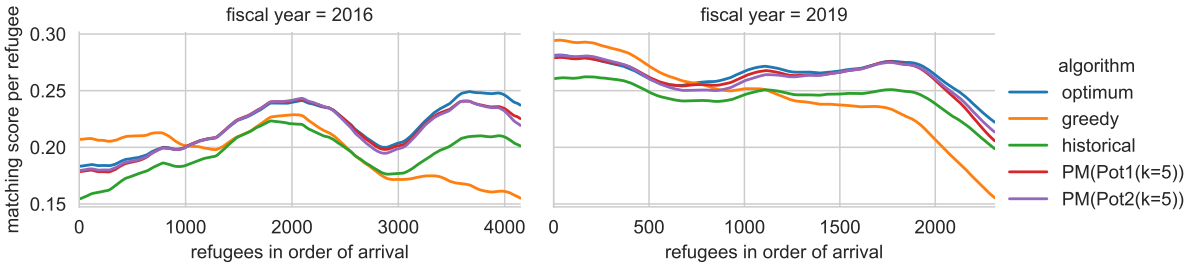


Figure I.2: Evolution of the per-refugee match score in order of arrival, for fiscal years 2016 and 2019 in the experiment of fig. I.1 (whole cases, final capacities). Match scores are smoothed using triangle smoothing with width 500.

a slider to indicate the expected number of arriving refugees, and could see how this would impact the recommended allocation of the current batch.

I.0.3 Employment statistics if capacities are changed during the year

For the fiscal years 2017 and 2018, we run a version of the experiments in section 2.7.1, in which the capacities are actually reduced at the time they were revised. For the greedy algorithm and the potential algorithms, this means that they are initially run using the initially announced capacities (and the potential algorithms expect 91% of the initial total capacity to arrive). At the time the capacity was revised, the capacity for the greedy algorithm and the potential algorithms is updated to the revised capacity. If the revised capacity of an affiliate lies below what the algorithm already used before the the point of revision, the capacity is frozen at current occupancy. As we argue in the body of the paper, this gives the greedy algorithm an unfair advantage and makes the resulting total employment hard to compare. The historical matching is roughly comparable since HIAS also started out aiming for the full capacities and later aimed for the revised capacities. However, the historical matching in 2017 actually made use of a fairly large end-of-year increase in capacity, which the other algorithms do not have access to, and it profits from the fact that HIAS influenced the change in capacity to match its

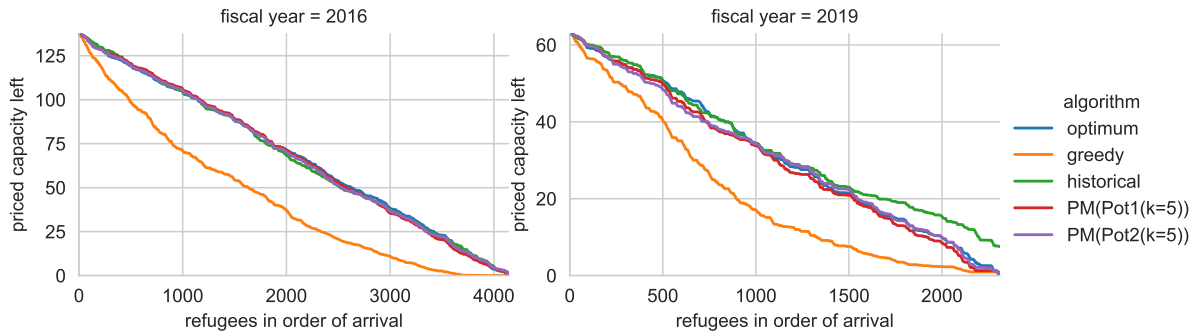


Figure I.3: Remaining priced capacity at the time of arrival of different refugees, for fiscal years 2016 and 2019 and one random run per algorithm in the experiment of fig. I.1 (whole cases, final capacities).

prior allocation decisions. As a reference, we show optimal matchings in hindsight both for the initial and for the revised capacities.

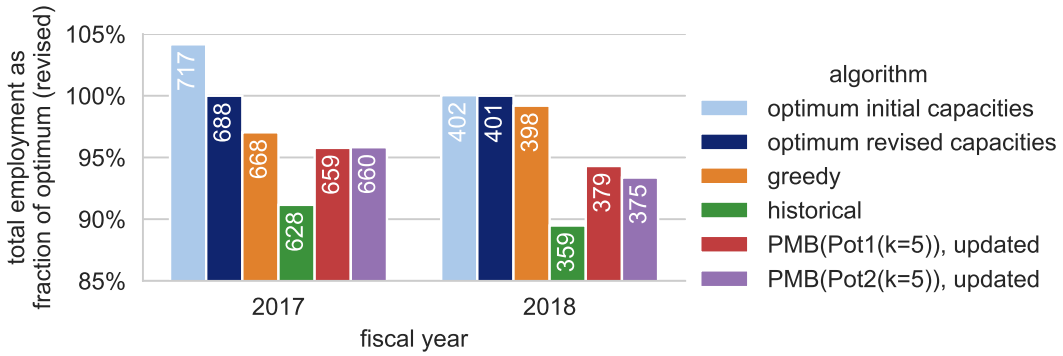


Figure I.4: Total employment, where cases are not split up and arrive in batches. Dynamic algorithms start respecting initial capacities, but capacities are changed at time of revision (except for historical). The potential algorithms do not have access to the true number of arriving cases but assume that the arriving refugees amount to 91% of the initial, then the revised capacity. For potential algorithms, a single random run is shown.

I.0.4 Percentage of Matched Refugees

Here, we report the percentage of refugees (i.e., cases weighted by case size) that gets matched to a real affiliate rather than the unmatched affiliate $\perp$. In our implementation of PMB (algorithm 3), for a small constant ϵ, we add a weight of $\epsilon \cdot s_i$ for all variables $x_{i,\ell}$ indicating that a case i is matched to an affiliate $\ell \neq \perp$, with the intention of breaking ties between optimal solutions in favor of solutions that don't leave more cases unmatched than necessary. This is particularly relevant since some cases, for example those consisting of unaccompanied minors, have an employment score of zero for all affiliates. This added constant (and the equivalent modification to PM) ensures that whether these cases ever get matched does not depend on the implementation of the ILP solver. Since we implement the greedy algorithm by setting potentials in PMB to zero, the same holds true for greedy. Finally, a similar issue

arises for the optimum matching in hindsight. For this, we optimize total employment without further constraints on the number of matched refugees, but then greedily add cases with zero unemployment where they fit throughout the year.

In particular, this means that the optimum is not an upper bound on how many people are matched. Instead, it is the historical matching that matches 100% of refugees, but this comparison is not on equal terms since the historical matching conforms to final capacities that were specifically increased to fit all refugees and since it ignores some case–affiliate incompatibilities.

For the setting of online bipartite matching, we find in fig. I.5 that the potential algorithms are roughly on par with the greedy algorithm when it comes to matching many refugees. The fact that both greedy and potential algorithms tend to leave a single-digit percentage of refugees unmatched can probably be explained by the many cases with family ties, which can only go in a single affiliate and have to remain unmatched if this affiliate is full, which is hard to avoid without knowing future arrivals. The optimum-hindsight matching does not have this problem, and therefore matches around 2% of refugees more on average. The remaining gap to 100% can be explained by the greedy matching of zero-employment cases, by the fact that the case–affiliate incompatibilities might not allow to match all cases (which the historical matching is not bound by), and by genuine trade-offs between matching more refugees and obtaining higher total employment.

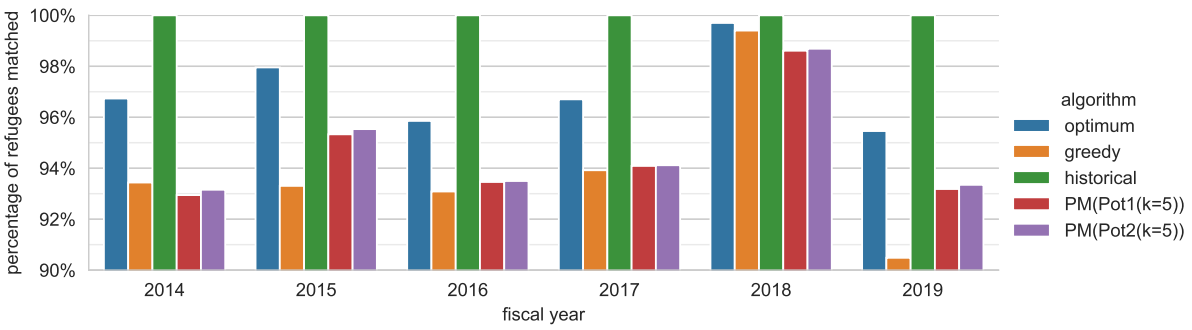


Figure I.5: Match percentages for the experiment in fig. 2.1 (split cases, final capacities).

As for the total employment, the addition of non-unit case sizes (fig. I.6) and of batching (fig. I.7) barely has any effect on the percentage of matched refugees.

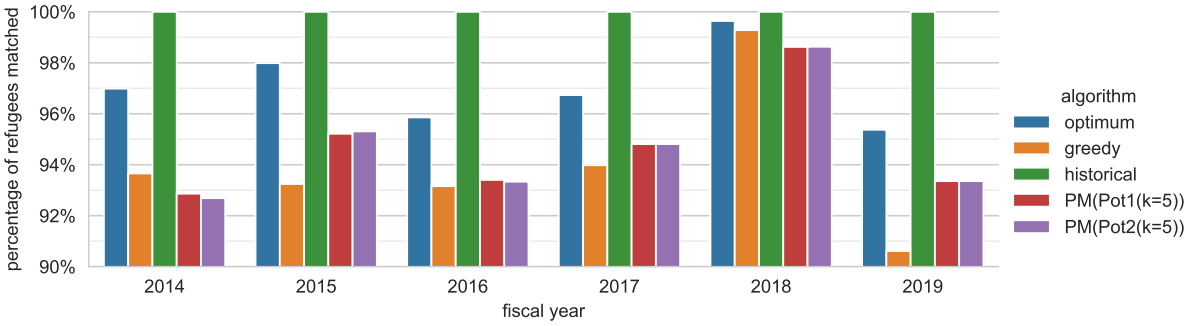


Figure I.6: Match percentages for the experiment in fig. I.1 (whole cases, final capacities).

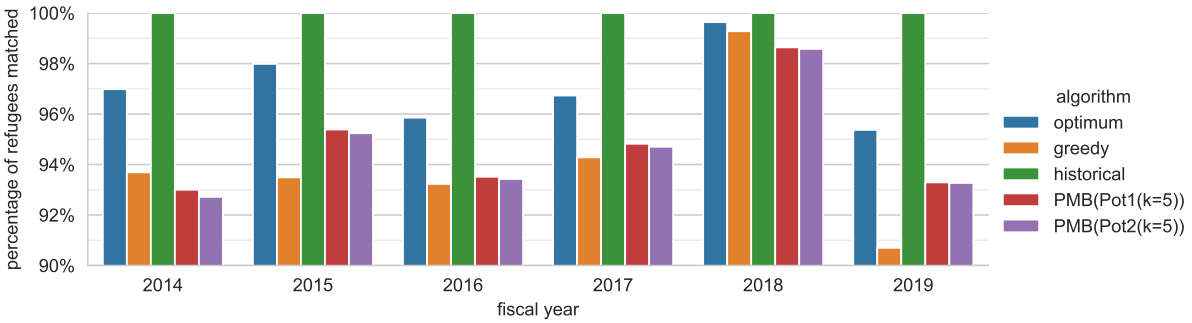


Figure I.7: Match percentages for the experiment in fig. 2.4 (whole cases, batching, final capacities).

In fig. I.8, the percentage of matched refugees by the optimum matching and the greedy algorithm is lower in most years, which is due to the initial capacities being strictly tighter than the final capacities on all years other than 2017 and 2018. The potential algorithms follow the same trend, but visibly match particularly few refugees in fiscal years 2017 and 2018. Since these algorithms vastly overestimate the arrival numbers due to the overly large capacities, they leave some refugees unmatched in the expectation that this will benefit (spurious) later arrivals.

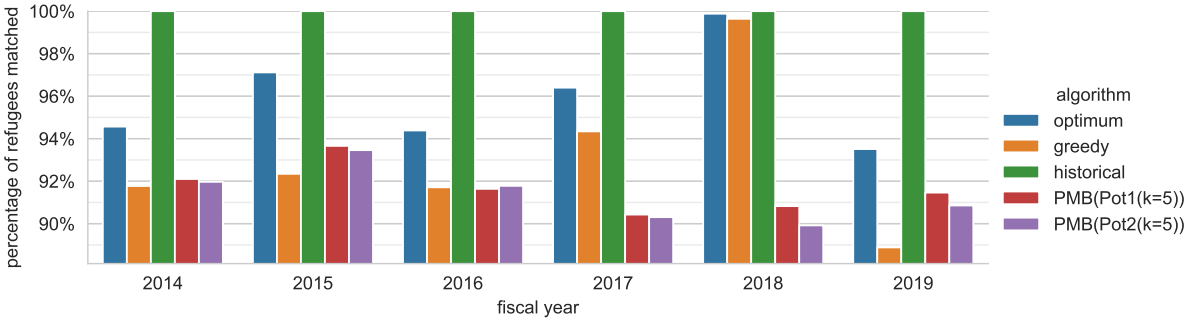


Figure I.8: Match percentages for the experiment in fig. 2.6 (whole cases, batches, initial capacities, potential algorithms do not know n).

Appendix J

Chapter 3: Collective Risk, versus Individual Risk, Simple Example

Comparing mathematical formulations in INDIVIDUAL-RARR and COLLECTIVE-RARR and analyzing the outcomes of these two models reveals interesting results. While COLLECTIVE-RARR hedges against the risk according to the chosen value of ρ, the focus of this mean-risk formulations is on the risk around the total expected employment, and it fails to adequately alleviate the risk at the refugee family level. Thus, the variation around expected outcome—mean employment likelihood—of each refugee family at its optimal placement is not necessarily reduced by this formulation. We illustrate these relationships in a form of a simple example.

In this example we have two refugee families, two affiliates with enough capacity to place both refugee families, and two scenarios in which we estimate the employment probability of families in affiliates. Let $[x_1^1, x_2^1, x_1^2, x_2^2]$ be a list of placement decisions where the first two items belongs to placement decisions of first family into affiliates one and two respectively, and the second two items belongs to placement decisions of second family into affiliates one and two respectively. The placement decision list can take following feasible solutions: $[1, 0, 1, 0], [1, 0, 0, 1], [0, 1, 1, 0], [0, 1, 0, 1]$ as each family can be placed in only one location. Let $[v_1^{11}, v_2^{11}, v_1^{21}, v_2^{21}] = [0.36, 0.68, 0.17, 0.28]$ be a list of estimated employment for families in scenario one, and $[v_1^{12}, v_2^{12}, v_1^{22}, v_2^{22}] = [0.64, 0.18, 0.97, 0.78]$ be a list of estimated employment for families in scenario two. Consequently, we have the mean estimates list $[\bar{v}_1^1, \bar{v}_2^1, \bar{v}_1^2, \bar{v}_2^2] = [0.5, 0.43, 0.57, 0.53]$, (i.e. $\bar{v}_1^1 = (v_1^{11} + v_1^{12})/2$). If the first family is placed in the first affiliate, the expected employment would be 0.5 and the variation around this expected value would be ± 0.14, if either scenarios are realized. The expected employment and variation around the expected would be 0.43 and ± 0.25 for this family in second location. So we say the decision about placing first family in first location is less risky than placing it in the second location as there is less variation around the mean estimate. Similarly, for the second family, the expected employment and variation around it is 0.57, ± 0.4 in first location and 0.53, ± 0.25 in second location, respectively. Thus, the first location is more risky for second family. Therefore, we have four combination of total risk, observed by INDIVIDUAL-RARR. First, when we place both refugees in less risky locations and the total risk is: $(w^1 |\, x_1^1 = 1) + (w^2 |\, x_2^2 = 1) = 0.0821$. Second, when the first family is placed in more risky option (location two) and second family is placed in less risky option (location two) where the total risk is: $(w^1 |\, x_2^1 = 1) + (w^2 |\, x_2^2 = 1) = 0.125$. Third case is the opposite of second case and total risk is: $(w^1 |\, x_1^1 = 1) + (w^2 |\, x_1^2 = 1) = 0.1796$. Finally, when both families are placed in

the more risky locations and the total risk is: $(w^1|\ x_2^1 = 1) + (w^2|\ x_1^2 = 1) = 0.2225$. Table J.1 shows the numerical details of this example for each feasible solution. From the point of view of Collective-RARR, the risk of each feasible solution is defined differently and is calculated as the variance of total employments in two scenarios around the total expected employment shown in the last column of Table J.1.

Table J.1: caption

Feasible Decision	Total Expected Employment $\bar{q}$	Individual-Level Risk observed by Model (3.3)	Individual-Level Risk Rank	Collective-Level Risk observed by Model (3.2)
$[1, 0, 1, 0]$	1.07	0.1796	3	0.2916
$[1, 0, 0, 1]$	1.03	0.0821	1	0.1521
$[0, 1, 1, 0]$	1.00	0.2225	4	0.0225
$[0, 1, 0, 1]$	0.96	0.125	2	0.0

When we optimally place families with Individual-RARR, the optimal solution is obtained based on the trade-off in the objective between the total expected employment shown in second column and the individual-level risk in third column weighted by parameter ρ. Therefore the optimal solution changes among the four feasible decision when the ρ is changed. However for Collective-RARR, the trade-off in the objective is with respect to collective-level risk shown in the last column. Thus the change in optimal solution obtained by changing the risk parameter would be different for these two models as they are penalised with different risk values.

Figure J.1 depicts the change in objective function and individual-level risk rank of optimal solution for these two models. The circle line shows the objective value (3.3a) and triangle line shows the objective value (3.2a) when ρ increases on horizontal axis. These lines are color-coded based on the individual-level risk rank of the optimal solution where the light blue belongs to the risk rank 1 when both families are placed in their low risky locations and dark blue belong to the risk rank 4 when both families are placed in their high risky locations and so on. When $\rho = 0$ the results is the same for both models and the optimal solution is the first one with risk rank 3. For non-zero values of ρ both objectives are penalized but by different risk measures. By increasing the ρ, at some point, both models switch to different optimal placement decision imposed by the conditions for trade-off. Collective-RARR only cares about the total employment and risk around it and ignores the family-level risk, thus it places families in risk rank 4 and later in risk rank 2. On the other hands, Individual-RARR consider the family-level risk and places families in risk rank 1. Consequently Individual-RARR is able to hedge against the risk in family level while Collective-RARR is indifferent to it.

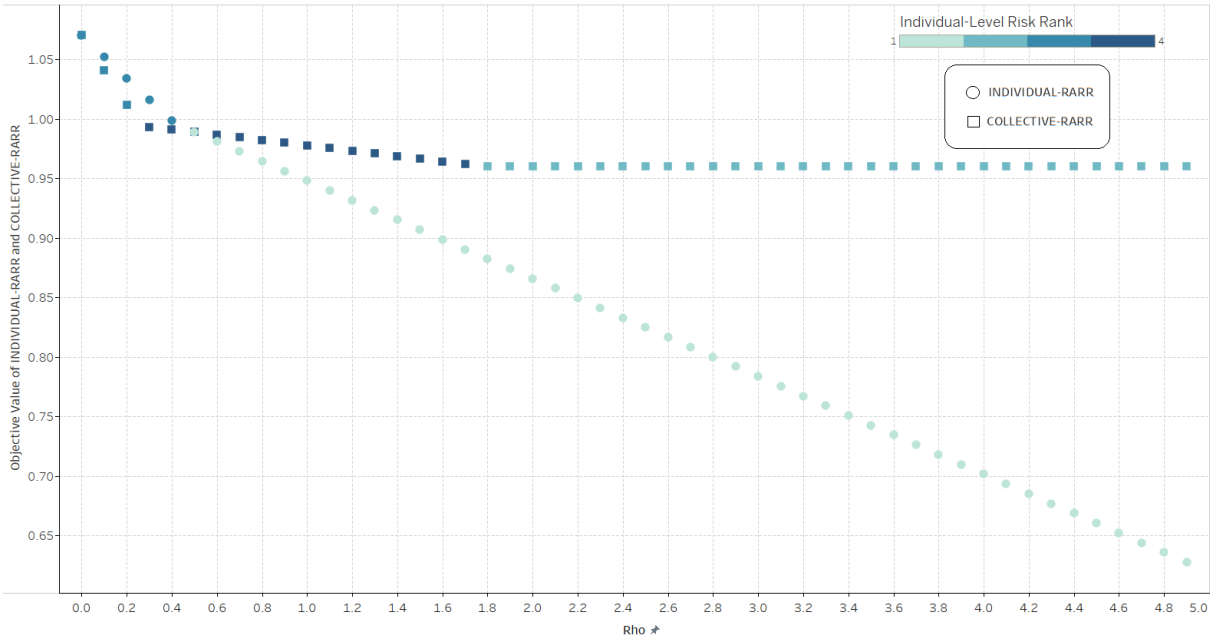
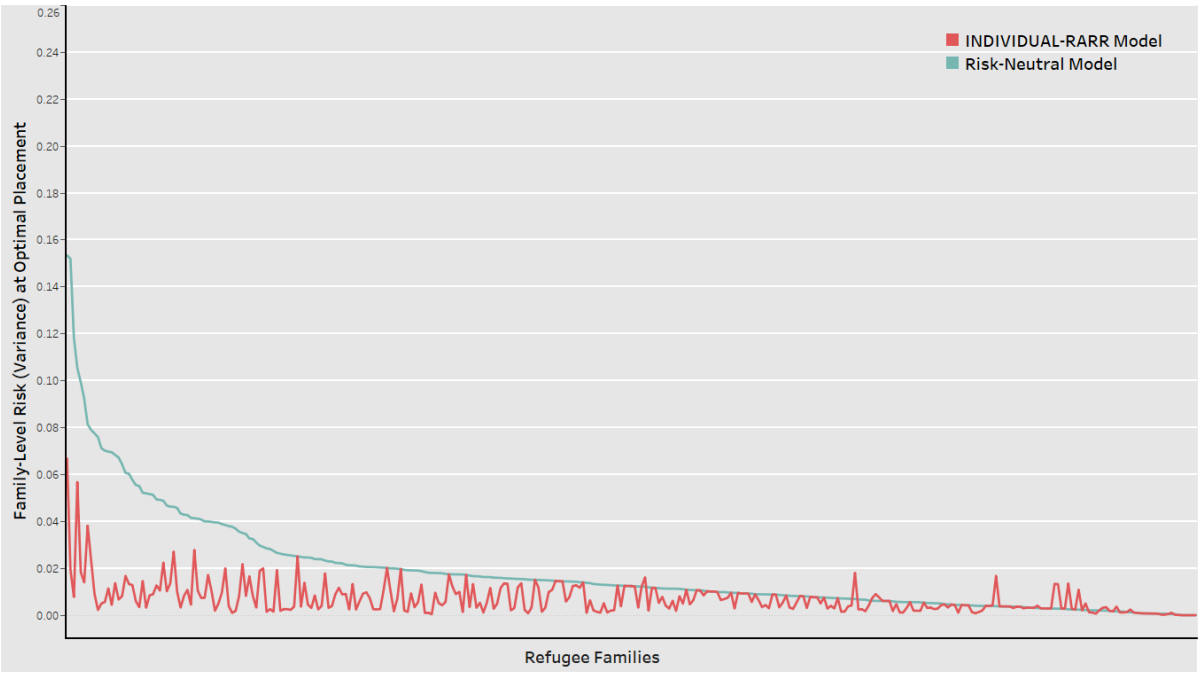


Figure J.1: Change in objective function value (3.2a) in COLLECTIVE-RARR and (3.3a) in INDIVIDUAL-RARR when ρ is increased. Points are color-coded and correspond to different family-level risk rank.

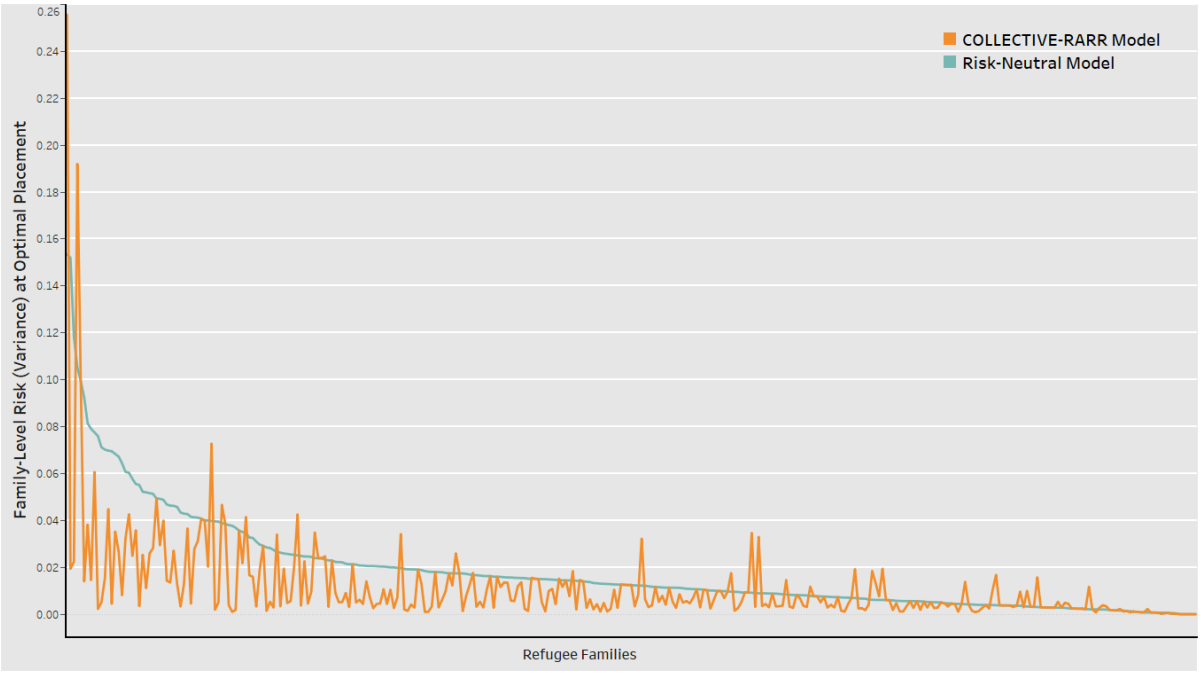
Appendix K

Chapter 3: Collective Risk, versus Individual Risk, Families' Risk Visualization

We solve the Individual-RARR and Collective-RARR on real refugee data including 329 refugee families (839 refugees) arrived in 2017 at HIAS. The Figure K.1 depicts the results. The horizontal axes on both K.1a and K.1b show the refugee families and the vertical axes show the family-level risk formulated in (3.4). The blue lines in both graphs show the risk at the optimal placements at risk-neutral setting ($\rho = 0$). Both graphs are sorted based on values on the blue lines. The red line in Figure K.1a shows the risk at optimal placements under Individual-RARR at $\rho = 45$ where loss on total expected employment is about $\%21$ of the total optimal employment at risk-neutral setting. The orange line in Figure K.1b shows the risk at optimal placements under Collective-RARR at $\rho = 5$ where loss on total expected employment is about $\%21$ of the total optimal employment at risk-neutral setting. In both graphs, the data points that are below the blue line are families that the corresponding risk-averse model placed them in less risky location compared to risk-neutral setting. The data points that are above the blue line are families that corresponding risk-averse model placed them in more risky location compared to risk-neutral setting. We can see that the number of families below the blue line in first graph is much more than families below the line in second graph. Also the magnitude of risk reduction for families in first graph is much higher than second graph. This demonstrates that the Individual-RARR performs better on family-level risk reduction compare to Collective-RARR.



(a) Risk of refugee families formulated in (3.4), by solving INDIVIDUAL-RARR. Blue: risk-neutral setting ($\rho = 0$). Red: risk-averse setting ($\rho = 45$ and loss on total expected employment is about %21 of the total optimal employment of risk-neutral setting).



(b) Risk of refugee families formulated in (3.4), by solving COLLECTIVE-RARR. Blue: risk-neutral setting ($\rho = 0$). Orange: risk-averse setting ($\rho = 5$ and loss on total expected employment is about %21 of the total optimal employment of risk-neutral setting).

Figure K.1: Family-level risk for refugee families. INDIVIDUAL-RARR performs better on family-level risk reduction compare to COLLECTIVE-RARR.