Combinatorial Discrete Choice*

Costas Arkolakis[†] Fabian Eckert[‡] Rowan Shi[§]

First Version: January 2017

This Version: April 2021

We study combinatorial discrete choice problems in which an agent chooses items from a set and the return to each item depends on which others are chosen. We develop an approach to solve such decision problems and to aggregate optimal decisions across heterogeneous agents. In order to solve a single agent's decision problem, we impose a restriction on the complementarities between items. To aggregate across heterogeneous agents requires an additional restriction on the complementarities between items and agent type. We argue that both restrictions naturally arise in many economic models. We present a generalized treatment of plant location and multi-stage input sourcing problems, highlighting the role that our restrictions play to solve and aggregate individual agents' problems.

*This paper combines "Combinatorial Discrete Choice" by Arkolakis and Eckert, and the technical part of the first chapter of Shi's dissertation titled "Solving combinatorial discrete choice problems in heterogeneous agent models: theory and an application to corporate tax harmonization in the European Union." We thank Jake DiCicco, Tra Nguyen, Andrew Salmon, Benjamin Tong and Yunus Tuncbilek for outstanding research assistance. We also thank Chrysanthos Gounaris for early conversations on the topic as well as David Atkin, Arnaud Costinot, Dave Donaldson, Penny Goldberg, Navin Kartik, Samuel Kortum, Eduardo Morales and Steve Redding for comments and suggestions. Code implementing the techniques introduced in this paper can be found at https://github.com/rowanxshi/cdc_solver.

[†]Yale University & NBER; costas.arkolakis@yale.edu

[‡]University of California, San Diego; fpe@ucsd.edu

[§]Ryerson University; rowanxshi@ryerson.ca

1. Introduction

"Combinatorial discrete choice problems" (CDCPs) in which agents choose a set of items (e.g., locations) and the return to each item depends on which others are chosen are ubiquitous in economics. For example, multinational firms like Volkswagen have to decide on a set of countries in which to operate production plants to serve world markets. For each destination market, Volkswagen then chooses a plant to minimize shipping costs, making plants substitutes. Similarly, large companies like Boeing often have to decide on a set of countries with which to establishing input sourcing relationship. Depending on the context, the different source it chooses could be complements or substitutes. Such CDCPs are hard since the number of potential decision sets grows exponentially in the number of available items. As a result, most existing work either avoids to confront these problems directly or limits the number of items (e.g., locations) to choose from. Aggregating such decision problems across many heterogeneous agents is even more difficult since the optimal decision set can vary arbitrarily with agent type.

We develop a new approach to solve such discrete choice problems with interdependencies and aggregate optimal decisions across arbitrary distributions of agent heterogeneity. Our approach relies on two restrictions on agents' return functions that are commonly satisfied in economic models. The first restriction constrains the complementarities between individual choices in an agent's return function, the second how decisions interact with agent type. The first restriction alone gives rise to an iterative technique to solve discrete choice problems with interdependencies. The combined restrictions allow to solve for a policy function that maps agent types to optimal decision sets in settings with heterogeneity. Aggregate outcomes result from integrating the policy function over the agent type distribution. Finally, we show that in many economic contexts our restrictions are automatically satisfied as a by-product of the economic structure of the problem. One example are the plant location choice and input choice problems that arise in the literature. We thus discuss how to use our methodology to solve and aggregate two generalized formulations of these problems.

We start by defining a general class of problems that we term "combinatorial discrete choice problems," or CDCPs for short. In these problems, an agent maximizes a return function π by choosing items from a discrete "choice set" J. Solving the CDCP means identifying the agent's *optimal* decision set $\mathcal{J}^\star \subseteq J$ which maximizes the return function π. Depending on the application, an agent could be a firm or consumer and the set J a list of potential store locations, export destinations, or items to purchase in the supermarket. These problems differ from the classical discrete choice problems in which each agent chooses one alternative from a list of mutually exclusive options. In heterogeneous agent models, return functions may differ by agent type. In such settings, we are interested in the *policy function*, $\mathcal{J}^\star(\cdot)$, mapping agent type to optimal decision set. The policy function summarizes the optimal behavior of all agents and, when aggregated over the type distribution, determines aggregate equilibrium outcomes.

The first part of this paper provides a method to solve a single agent's CDCP. We show how imposing a simple restriction on an agent's return function — *single crossing differences in choices* (SCD-C) — gives rise to an intuitive solution method.[1] There are two types of SCD-C conditions: SCD-C from above and below. Our method applies as long as an agent's return function satisfies either of the two SCD conditions. SCD-C from above implies that, if including an additional item yields a positive value given some other chosen items, it continues to do so as other items are removed. Likewise, SCD-C from below implies that, if including an additional item yields a positive value given some other chosen items, it continues to do so as other items are added. SCD-C hence restricts the strength of complementarities between different decisions. Supermodularity and submodularity of the return function are sufficient conditions for SCD-C from below and above respectively.[2]

We solve CDCPs with return functions that exhibit SCD-C by iteratively eliminating non-optimal decisions sets. Intuitively, by evaluating the return function at points of extreme complementarities, potential solutions can be discarded without having to evaluate them. For example, with positive complementarities, we know that items that reduce the value of the return function when added to a decision set that includes all other items, are not part of the optimal decision set. Likewise, with negative complementarities, items that increase the return function even when no other items are chosen must be part of the optimal decision set since they add value even without the help of positive complementarities through other items. Applying this logic iteratively shrinks the number of potential decisions sets without ever discarding the optimal decision set $\mathcal{J}^\star$. When no more decision sets can be eliminated, we are either left with the single optimal decision set, or multiple decision sets among them the optimal one. For the latter case, we define an additional step that often finds the optimal decision set without evaluating the return function for all remaining ones.

To set ideas, consider a classical plant location problem of a firm choosing optimal locations for its plants or retail outlets. Locations without a plant are served from location with a plant via trade. As a result, a firm's individual plants are usually substitutes, so that SCD-C from above holds and our method is applicable. With N alternative discrete locations, the firm's choice space contains 2^N potential plant strategies. Without our method researchers would have to calculate the return

[1]The importance of single crossing differences in comparative statics analysis in mechanism design has been widely discussed (see Milgrom (2004)).

[2]Supermodularity and submodularity are of wide theoretical and practical use in economics in all contexts in which complementarities between choices are important. Supermodularity forces spillovers between decisions to always be positive, instead SCD-C from below allows for spillovers to be positive *or* negative, but restricts the strength of negative spillovers. Likewise submodularity implies that spillovers between decisions are always negative, while SCD-C from above only restricts them from being too positive.

to each strategy and choose the return-function-maximizing one, which may be computationally infeasible.

For settings with heterogeneous agents, we show how to solve for the policy function mapping agent type to optimal decision sets without solving every agent's CDCP separately or relying on approximations. To do so, we impose a second restriction on agents' return functions: single crossing differences in type (SCD-T). SCD-T guarantees that for any potential decision set and any item, the typespace can be split into two contiguous groups: those that would derive a positive marginal value from including the item and others that would not. Intuitively, SCD-T guarantees that similar agents have identical optimal decision set so that the policy function only changes value at discrete points in the typespace. We solve for the policy function by identifying these points and the optimal decision sets shared by the types between them.[3]

We use the same logic as before to eliminate non-optimal decision sets for subgroups of types, but introduce an additional step to iteratively refine the partitioning of the typespace. For a given subgroup of types, the SCD-T assumption guarantees the existence of a cutoff type that divides the group into types for which a given decision set can be eliminated and those for whom it cannot. So at each iteration, we both eliminate decisions sets and further partition the typespace. Repeating this process yields an increasingly finer partitioning of the typespace, each with more and more non-optimal decision sets eliminated. The method converges once we can neither further reduce the choice space for a given subgroup of types nor refine the typespace partitioning. For groups of types for which a single decision set remains in the choice space, we have found the optimal decision set, and hence the value of the policy function over that region of the typespace. For groups with multiple potential decision sets remaining, we introduce a second step and prove that applying it recursively always yields regions of the typespace with the associated optimal decision.

The third part of our paper provides a generalized treatment of the plant location and multi-stage input sourcing problems that arise in the international trade and operations research literature, and provides general restrictions under which our techniques can be applied to solve them. We define a general class of plant location problems and prove that the economic structure of these problems itself implies that locations are substitutes. As a result, the entire class of these problems always satisfies SCD-C from above so that our methods can be used to solve the individual firm problem. We additionally provide easy-to-check sufficient conditions for SCD-T to be satisfied. We also define a class of input sourcing problems, including problems with multiple stages of production often referred to as (global) value chain problems (see e.g., Antràs and Chor (2021)). Such problems contain forces that make individual sourcing location both complements and substitutes We provide

[3] The policy function can exhibit jumps, non-monotonicities, and partially overlapping decisions sets. Our approach is applicable for continuous and degenerate, single and multidimensional type distributions alike.

specific restrictions that imply that one set of forces dominates the other so that either SCD-C from below or above holds and our methods apply. We also discuss sufficient conditions on preferences, technology, and agent heterogeneity for SCD-T to holds so that aggregation across heterogeneous agents is feasible. We finally argue that our generalized plant location and multi-stage input sourcing frameworks nest existing treatments and discuss possible extensions that the flexibility of our framework allows.

Our paper contributes to two distinct literatures. First, we add to a rapidly growing literature in international trade and industrial organization in which firms choose a discrete set of locations to build plants in or source inputs from.[4] These problems are quintessential CDCPs since the return to one location depends on which others are chosen. Most of the existing literature has either made these CDCPs trivial by abstracting from the fixed cost of adding a location (see Ramondo (2014), Ramondo and Rodríguez-Clare (2013), Arkolakis et al. (2018)) or solved them only for a small number of discrete locations so that it is feasible to evaluate the return to all possible combinations of location decisions separately (see Tintelnot (2016) and Zheng (2016)).[5] Other papers have specified the CDCP and instead of solving it, estimated its parameters using moment inequalities (see Pakes et al. (2015), Holmes (2011), or Morales et al. (2019)); while this approach generates insights about the parameters of the firm problem it does not allow for counterfactual analysis, which would require solving the firms' CDCP explicitly. Jia (2008) presents a notable exception in introducing a "reduction method" that helps solve CDCPs with fixed costs when the return function is supermodular (see also Antras et al. (2017)). Our paper adds a new and unified way of solving CDCPs with positive or negative complementarities to this literature. Relative to Jia (2008), we present a generalized method applicable to return functions exhibiting a weaker form of positive spillovers, or, more importantly, negative spillovers. As explained above, such negative complementarities are an inherent feature of plant locations problems.[6] We also introduce a new method to aggregate combinatorial discrete choices across heterogeneous agents.

The third part of our paper further contributes to the same literature by introducing a general class of plant location problems that nests those many canonical papers (Balinski (1965), Owen and Daskin (1998), Tintelnot (2016), Arkolakis et al. (2018), Ramondo (2014), and Ramondo and

[4]The classic Simple Plant Location Problem in operations research is an NP-hard problem that likewise fits the description in 2 (see Jakob and Pruzan (1983)).

[5]Without fixed costs, firms can trivially choose to include all locations in their choice set, even if some remain idle. Tintelnot (2016) uses 12 countries which allows him to compute the firm profits from any possible combination of plant locations of which there are 2^{12}. Zheng (2016) partitions the United States into small regional markets for which plant location decisions interact; decisions across these regional markets are independent, within them there a few enough locations to follow the "brute force" approach of Tintelnot (2016). The brute force approach becomes impossible with about 20 locations in most applications.

[6]See Yang (2020) for a recent application of our method to study the location choices of multi-plant oligopolists in the cement industry, a CDCP with negative complementarities.

Rodríguez-Clare (2013)). Similarly, we present a generalized international sourcing problem that nests existing treatments of both multi-stage global value chain problems (as in Antràs and de Gortari (2017)) and single-stage sourcing problems as in Antras et al. (2017). Our treatment provides a unified perspective on these problems and easy-to-check conditions to determine whether any given problem can be solved using the techniques introduced in this paper.

A second distinct contribution is our method for solving for the policy function mapping agent type to optimal decision, which together with the agent type distribution can be used to aggregate decisions. Random utility approaches (see McFadden (1973) or Eaton and Kortum (2002)) add random shocks to individual choices and assume them to be extreme value distributed which yields analytical expressions for the fraction of agents choosing each discrete alternative. However, these classical discrete choice models allow for one choice among mutually exclusive alternatives only (see Guadagni and Little (1983) and Train (1986)), so complementarities between different choices are not relevant by assumption.[7] As a result, standard discrete choice tools with random utilities are then not useful to study CDCPs.[8] Thus far, the literature has hence aggregated the solution to CDCPs solved by individual agents by discretizing the typespace and solving the CDCP only for a limited number of types, interpolating in-between them. We show that that such interpolation can lead to large errors in setting with negative complementarities since the policy function exhibits no form of continuity or nesting. We add to this literature by providing a method to solve for the policy function that is exact, without the need for approximation, and usually faster since we directly solve for all "kinks" in the policy function where the optimal decision set changes.

2. Combinatorial Discrete Choice Problems

We consider an economy inhabited by a set of heterogeneous agents indexed by their type $\mathbf{z} \in \mathbf{Z} \subseteq \mathbb{R}^N$. The "aggregate state" of the economy is denoted by $\mathbf{y} \in \mathbf{Y} \subseteq \mathbb{R}^M$. Agents choose a subset $\mathcal{J}$ of items from a finite discrete set J to maximize a return function of the form

$$\pi(\mathcal{J}; \mathbf{z}, \mathbf{y}) : \mathscr{P}(J) \times \mathbf{Z} \times \mathbf{Y} \to \mathbb{R}, \tag{1}$$

[7] Note that there are random utility type models with correlated shocks across choices, e.g., Bryan and Morten (2019), Arkolakis et al. (2018), and Lind and Ramondo (2018). However, in these settings individual agents continue to choose just one alternative, but in the *aggregate* certain choices may be correlated, e.g., countries purchasing a lot from Germany may also purchase a lot from neighboring Austria, not for random reasons but due to a correlations of unobserved tastes or productivities.

[8] Hendel (1999) provides a notable exception by extending the random utility approach to multiple discrete choices, however, continues to abstract from interdependence between an agent's decisions.

where the power set $\mathscr{P}(J) = \{\mathcal{J} \mid \mathcal{J} \subseteq J\}$ is the collection of all possible subsets of J. We refer to $\mathcal{J}$ as an agent's "decision set," to J as their "choice set," and to $\mathscr{P}(J)$ as their "choice space."

The formulation in equation 1 is very general. The return function could be a firm's profit function, and $\mathcal{J}$ the set of countries in which it chooses to operate a plant, or it could be an individual's utility function, and J the collection of items chosen in the supermarket. In many applications other continuous variables are chosen conditional on the choice of a discrete set $\mathcal{J}$ of items, e.g., conditional on having a plant in location A how much should this plant produce? In such situations, the return function can be expressed as in equation 1 after "maximizing out" the continuous choice variable.[9]

Agents' return to choosing item j may depend on what other items are included in $\mathcal{J}$, rendering the decision problem *combinatorial* in nature. To formalize the interdependence of decisions, we introduce a marginal value operator that encodes the additional benefit agents derive from element j's inclusion in the decision set $\mathcal{J}$.

Definition 1 (Marginal value operator)**.** *For an item $j \in J$, decision set $\mathcal{J}$, agent type $\mathbf{z}$, and aggregate vector $\mathbf{y}$, the marginal value operator D_j on the return function $\pi(\mathcal{J};\mathbf{z},\mathbf{y})$ is defined as*

$$D_j\pi(\mathcal{J};\mathbf{z},\mathbf{y}) \equiv \pi(\mathcal{J} \cup \{j\};\mathbf{z},\mathbf{y}) - \pi(\mathcal{J} \setminus \{j\};\mathbf{z},\mathbf{y}) \,. \tag{2}$$

Decisions are *interdependent* as long as the marginal value $D_j\pi(\mathcal{J};\mathbf{z},\mathbf{y})$ of item j depends on the overall choice set $\mathcal{J}$. If the marginal value of any item j is the same across all decision sets $\mathcal{J}$, decisions are independent and the agent can simply consider each item in isolation.

We now formally define the class of combinatorial discrete choice problems.

Definition 2 (Combinatorial discrete choice problem)**.** *A combinatorial discrete choice problem (CDCP) of an agent with type $\mathbf{z}$, given an aggregate state $\mathbf{y}$, is to find the optimal decision set $\mathcal{J}^\star$ such that*

$$\mathcal{J}^\star = \underset{\mathcal{J} \subseteq J}{\arg\max}\, \pi(\mathcal{J};\mathbf{z},\mathbf{y})$$

where the return function $\pi(\mathcal{J};\mathbf{z},\mathbf{y})$ is defined in equation (1).

Combinatorial Discrete Choice problems appear in many applications in economics. Their central feature is that individual agents choose potentially more than one item from a set of items, and the

[9] Dynamic combinatorial discrete choice problems can be written as in equation 1. Suppose each period a firm can choose in which additional locations to open plants. Equation 1 would then refer to the static problem of choosing the optimal additional plants in a given period, given the plants already present (which have to be removed from the choice set J) and given the future paths of plant decisions that would follow any decision today.

return to each item may depend on which others are chosen. They differ from classical discrete choice problems in which individual agents either choose one item from a set of mutually exclusive alternatives, or several items but without the return to each being a direct function of which others are chosen and with the total number of items to be chosen given as a parameter.

A classical CDCP is the plant location decision of a firm when its plants compete to serve demand points in each location (see Balinski (1965), Owen and Daskin (1998), Ramondo (2014), Ramondo and Rodríguez-Clare (2013), Arkolakis et al. (2018), Tintelnot (2016), Zheng (2016),Jia (2008), and Holmes (2011)). In these problems adding an additional plant in a given location may save on trade costs to serve the location's consumers, but takes away from the sales of another plant that used to serve this location via trade, making plant location decisions inherently interdependent. Other papers model instead the decision of firms with which countries to establish sourcing relationships (see Antras et al. (2017)), or outsourcing strategies in the presence of industry complementarities (see Jiang and Tyazhelnikov (2020)). In this setting, adding a new supplier reduces production costs, but reduces sales from each other supplier, lowering their marginal value, so that decisions are again interdependent. In other branches of economics researchers study the optimal evolution of networks, where one new link may beget others, or the optimal ATM or hospital network of locations, both can be formulated as CDCPs.[10]

In many of these applications, the economy is populated by a large number of agents whose return functions may differ. For such settings, we additionally define the set-valued policy function that summarizes the optimal decisions of all agents in the economy by mapping an agent's type to their optimal decision set, conditional on the aggregate state.

Definition 3 (Policy function)**.** *Consider a combinatorial discrete choice problem (CDCP) confronted by agents of heterogeneous types* $\mathbf{z}$ *and given an aggregate state* $\mathbf{y}$*. The policy function mapping agent type to an optimal decision set, conditional on the aggregate state* $\mathbf{y}$*, is defined as*

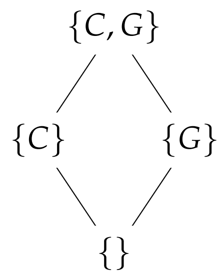
$$\mathcal{J}^{\star}(\cdot;\mathbf{y}) : \mathbf{Z} \rightarrow \mathscr{P}(J)$$

where the return function $\pi(\mathcal{J};\mathbf{z},\mathbf{y})$ *is defined in equation 1.*

A key aspect of the classic discrete choice literature is its ability to aggregate discrete choices across heterogeneous agents (see McFadden (1973)). To do so it adds random shocks to the return of each

[10]Economic models are often estimated using a method of moments that requires simulating agents' optimal decisions for different parameter values and computing moments. When agents' problems are CDCPs this has often proved impossible due to difficulty of solving them, especially repeatedly. Several papers in economics have hence resorted to estimating the parameters of a model that features a CDCP using moment inequalities (see Morales et al. (2019), Holmes (2011)). Our method opens the way to estimate these models via simulation, and also to conduct counterfactuals once parameters have been estimated.

FIGURE 1: THE FIRM'S POTENTIAL DECISION SETS



Notes: A two-country plant location problem. All four possible decision sets are represented. Lines indicate the subset structure of the decision space.

potential choice. Assuming that these shocks follow an extreme value distribution yields analytical expressions for the fraction of workers choosing each mutually exclusive alternative. The tools of these "random utility" models are much less useful when agents can choose an arbitrary subset of the available choices instead of just one alternative. As a result, most papers abstract from such situations and hence interdependencies between individual agents' choices. Our numerical approach differs from the extreme value aggregation in that it does not produce analytical expressions for the fraction of agents choosing each item. Its great advantage is that its designed for situation in which individual agents make interacting choices and works with arbitrary distributions of agent heterogeneity.

The next section proposes a method to solve an individual agent's CDCP, while Section 4 discusses recovering the exact policy function in settings with heterogeneous agents.

2.1. A Simple Example

We introduce a simple example, which we carry through the paper. Consider a profit-maximizing firm that chooses in which countries to establish a plant. There is some demand for the firm's product in each country, which can be served by shipping from a foreign plant or building a local plant. Suppose the available countries are Germany (G) and Canada (C). So the firm's choice set is $J = \{G, C\}$ and its choice space is $\mathscr{P}(J) = \{\{\}, \{G\}, \{C\}, \{G, C\}\}$, which contains all potential decision vectors of the firm.

Figure 1 represents all four possible decision sets available to the firm: it may open plants in both, just one, or none of the countries. Nested decision sets are linked with lines. Consider the marginal value of building a plant in Germany, $\mathcal{J} = \{G\}$, rather than no plant at all, $\mathcal{J} = \{\}$. What we learn from this marginal value depends on its sign and the type of complementarities between individual plants in the firm's return function.

For illustration, suppose the marginal value is negative and the return function features negative complementarities between plants. But then the negative marginal value implies that the optimal plant strategy does not involve a plant in Germany: having just a plant in Germany yields a negative return, adding another plant in Canada would further lower the marginal return to the German plant due to the negative complementarities.

The next section formalizes the idea of eliminating decision sets by studying the marginal value of individual decisions at points of maximum and minimum complementarity.

3. Solving CDCPs for a Single Agent

In this section, we show how to solve the CDCP of a *single* agent holding agent type $\mathbf{z}$ and the aggregate state $\mathbf{y}$ fixed. We introduce a restriction on the complementarities between choices in the return function called single crossing differences in choices, or SCD-C for short. For return functions satisfying SCD-C, we present a simple mapping on the choice space associated with a CDCP whose fixed point corresponds to the agent's optimal decision set.

3.1. Single Crossing Differences in Choices

The single crossing differences property in choices is defined as follows.

Definition 4 (SCD-C)**.** *Consider a return function π as defined in equation (1). For a given type $\mathbf{z}$ and aggregate vector $\mathbf{y}$,*

- $\mathbf{C^a}$. *The return function obeys single crossing differences in choices from above if, for all elements $j \in J$ and decision sets $\mathcal{J}_1 \subset \mathcal{J}_2 \subseteq J$,*

$$D_j\pi\left(\mathcal{J}_2;\mathbf{z},\mathbf{y}\right) \geq 0 \qquad\qquad \Rightarrow \qquad\qquad D_j\pi\left(\mathcal{J}_1;\mathbf{z},\mathbf{y}\right) \geq 0\,.$$

- $\mathbf{C^b}$. *The return function obeys single crossing differences in choices from below if, for all elements $j \in J$ and decision sets $\mathcal{J}_1 \subset \mathcal{J}_2 \subseteq J$,*

$$D_j\pi\left(\mathcal{J}_1;\mathbf{z},\mathbf{y}\right) \geq 0 \qquad\qquad \Rightarrow \qquad\qquad D_j\pi\left(\mathcal{J}_2;\mathbf{z},\mathbf{y}\right) \geq 0\,.$$

These restrictions are intuitive. For return functions satisfying condition $\mathrm{C^a}$, if the marginal value of including an additional element j in a given decision set is positive, it remains so as elements are removed from the decision set. Similarly, under condition C^b, if the marginal value of including an additional element j in a given decision set is positive, it remains so as other elements are added to the decision set. For the rest of the paper, we will refer to return functions π that satisfy either condition C^a or C^b as "exhibiting SCD-C."

A simple sufficient condition for SCD-C is for the marginal value of decision j, for all $j \in J$, to be monotone in its first argument. In particular, given any two sets $\mathcal{J}_1 \subseteq \mathcal{J}_2$, if

$$D_j \pi(\mathcal{J}_1; \mathbf{z}, \mathbf{y}) \geq D_j \pi(\mathcal{J}_2; \mathbf{z}, \mathbf{y}) \quad \forall j \in J,$$

the return function π necessarily obeys SCD-C from above and we say it satisfies the monotone substitutes property. If the weak inequality is flipped, the return function π satisfies SCD-C from below and we say it satisfies the monotone complements property.

The more restrictive monotone complements and substitute properties correspond directly to the notion of positive and negative complementarities common in economics. In particular, the marginal value of return functions that exhibit monotone substitutes decreases as more items are added to the decision set. Similarly, for return functions exhibiting monotone complements, any element's marginal value increases as more items are added to the decision set. In our setting, the definition of monotone substitutes and complements coincides with that of submodularity and supermodularity, respectively.[11]

The "Single Crossing Differences" property first appeared in Milgrom (2004).[12] While well-known in the microeconomics literature, to our knowledge it has not been discussed in the context of solving combinatorial discrete choice problems. Return functions often exhibit single crossing differences as a consequence of natural economic assumptions. One of the most prominent examples is the canonical plant location problem in economics and operations research. Consider a set of discrete demand points which can either be served by paying a fixed cost to build a local plant, or by shipping goods from another location that has a plant. The firm needs to decide in what subset of locations to build a plant. In this case, individual plants act as substitutes, since adding an additional plant into a location necessarily reduces sales of some other plant which was serving the location via trade before (see, e.g., Tintelnot (2017)).

[11] If the choice set is not finite the notions do not coincide. In this case, sub- and supermodularity are implied by monotonicity in set but not vice versa. We provide these results in the Appendix.
[12] Kartik et al. (2019) extend the notion of single crossing differences to distributions.

3.2. The Squeezing Procedure

This section presents our "squeezing procedure," a method to solve CDCPs when the return function exhibits SCD-C. At the heart of the solution method is a set-valued mapping applied to the choice space associated with a CDCP. The iterative application of the mapping eliminates an increasing number of non-optimal decision sets from the choice space and its fixed point always contains the optimal decision set. Since the type vector, $\mathbf{z}$, and aggregate state, $\mathbf{y}$, are held fixed in this section, we omit them for notational brevity.

Consider the choice set J of the CDCP defined in equation (1). We introduce an associated pair of sets $[\underline{\mathcal{J}}, \overline{\mathcal{J}}]$, which we call the "bounding sets." We use these sets to keep track of items from J that are certain to be included in or excluded from the optimal decision set $\mathcal{J}^\star$. The "subset" $\underline{\mathcal{J}}$ *includes* all items in $\mathcal{J}$ we know to *be* in the optimal decision set. The "superset" $\overline{\mathcal{J}}$ *excludes* all items in $\mathcal{J}$ we know to *not be* in the optimal decision set. The set difference of $\overline{\mathcal{J}}$ and $\underline{\mathcal{J}}$, denoted $\overline{\mathcal{J}} \setminus \underline{\mathcal{J}}$, is the collection of items which *may* be in the optimal decision set. We refer to this group as "undetermined" items or elements. A natural starting point for our procedure is to set $\underline{\mathcal{J}} = \emptyset$ and $\overline{\mathcal{J}} = J$, so that $\overline{\mathcal{J}} \setminus \underline{\mathcal{J}} = J$, that is, all items in J are undetermined.

The central mapping of our squeezing procedure, which we call the "squeezing step," acts on the bounding sets $[\underline{\mathcal{J}}, \overline{\mathcal{J}}]$ associated with the return function π. To formalize the squeezing step, we introduce an auxiliary mapping

$$\Omega(\mathcal{J}) \equiv \{j \in J \mid D_j\pi(\mathcal{J}) > 0\}$$

which collects the items $j \in J$ that have a positive marginal value as part of a given decision set $\mathcal{J}$. We then define the squeezing step as follows.

Definition 5 (Squeezing step)**.** *Consider a CDCP and its associated bounding sets* $[\underline{\mathcal{J}}^{(k)}, \overline{\mathcal{J}}^{(k)}]$.

S^a^. *The mapping* S^a *is such that*

$$S^a([\underline{\mathcal{J}}^{(k)}, \overline{\mathcal{J}}^{(k)}]) \equiv [\Omega(\overline{\mathcal{J}}^{(k)}), \Omega(\underline{\mathcal{J}}^{(k)})] \equiv [\underline{\mathcal{J}}^{(k+1)}, \overline{\mathcal{J}}^{(k+1)}],$$

S^b^. *the mapping* S^b *is such that*

$$S^b([\underline{\mathcal{J}}^{(k)}, \overline{\mathcal{J}}^{(k)}]) \equiv [\Omega(\underline{\mathcal{J}}^{(k)}), \Omega(\overline{\mathcal{J}}^{(k)})] \equiv [\underline{\mathcal{J}}^{(k+1)}, \overline{\mathcal{J}}^{(k+1)}],$$

where k indicates the output of the kth application of the squeezing step.

If the underlying return function satisfies SCD-C, each application of the squeezing step adds elements to the subset, $\underline{\mathcal{J}}$, while removing elements from the superset, $\overline{\mathcal{J}}$, thereby eliminating some non-optimal decision sets from the choice space of the CDCP. Iteratively applying the squeezing step converges to a fixed point on the bounding sets in polynomial time. We establish both of these results in the following theorem.

Theorem 1. *Consider a CDCP as defined in equation (1).*

1. *If the return function exhibits SCD-C from above, then successivly applying S^a to $[\emptyset, J]$ returns a sequence of bounding sets where $\underline{\mathcal{J}}^{(k)} \subseteq \underline{\mathcal{J}}^{(k+1)} \subseteq \mathcal{J}^\star \subseteq \overline{\mathcal{J}}^{(k+1)} \subseteq \overline{\mathcal{J}}^{(k)}$.*
2. *If the return function exhibits SCD-C from below, then successivly applying S^b to $[\emptyset, J]$ returns a sequence of bounding sets where $\underline{\mathcal{J}}^{(k)} \subseteq \underline{\mathcal{J}}^{(k+1)} \subseteq \mathcal{J}^\star \subseteq \overline{\mathcal{J}}^{(k+1)} \subseteq \overline{\mathcal{J}}^{(k)}$.*
3. *Conditional on the appropriate SCD-C condition, iterating on the mapping S^a or S^b converges in $O(n)$ time.*

Proof. See Appendix. □

Theorem 1 ensures that applying the squeezing step (weakly) reduces the collection of undetermined items.[13] In particular, the expression $\underline{\mathcal{J}}^{(k)} \subseteq \underline{\mathcal{J}}^{(k+1)}$ implies that (weakly) more items are *included* in the subset — and hence known to be in the optimal decision set — after applying the squeezing step. Similarly, the expression $\overline{\mathcal{J}}^{(k+1)} \subseteq \overline{\mathcal{J}}^{(k)}$ implies that (weakly) more items are *excluded* from its superset — and hence known not to be in the optimal decision set — after applying the squeezing step. Crucially, no items that are in the optimal decision set are erroneously included or excluded, since $\underline{\mathcal{J}}^{(k+1)} \subseteq \mathcal{J}^\star \subseteq \overline{\mathcal{J}}^{(k+1)}$.

We denote the total number of iterations until convergence by K. Accordingly, we denote operators that indicates applying the mappings $\mathrm{S^a}$ and $\mathrm{S^b}$ until convergence by $\mathrm{S}^{\mathrm{a}(K)}$ and $\mathrm{S}^{\mathrm{b}(K)}$ and by $[\underline{\mathcal{J}}^{(K)}, \overline{\mathcal{J}}^{(K)}]$ the resulting bounding sets. In the Appendix, we establish that K is never larger than the cardinality of the choice set, $|J|$.

Consider the bounding sets resulting from applying $\mathrm{S^a}$ or $\mathrm{S^b}$ until convergence. If the converged pair of bounding sets is identical such that $\underline{\mathcal{J}}^{(K)} = \overline{\mathcal{J}}^{(K)}$, Theorem 1 implies that $\mathcal{J}^\star = \underline{\mathcal{J}}^{(K)} = \overline{\mathcal{J}}^{(K)}$ so that we have identified the optimal decision set solving the CDCP. The next subsection shows how to identify the optimal decision set when the converged pair of bounding sets is not identical.

A limitation of our approach is that the return function has to satisfy the same type of SCD-C (i.e.,

[13] The squeezing step is designed to recover $\mathcal{J}^\star$ so that all items j for which the agent is indifferent are excluded from the optimal strategy. If these items should be included, they are easily identified as those j for which $D_j\pi(\mathcal{J}^\star) = 0$.

either below or above) over the entire choice space.[14]

3.3. The Branching Procedure

Sometimes we are left with several potentially optimal decision sets after convergence of the squeezing procedure. One option is then to apply the computationally expensive brute force method of evaluating the return function at all remaining decision sets. In this section, we introduce a "branching procedure" that often finds the optimal decision set much faster than brute force.

At the heart of the branching procedure is a "branching step" applied to a CDCP for which the squeezing procedure has converged. The branching step takes an undetermined item j such that $j \in \overline{\mathcal{J}}^{(K)} \setminus \underline{\mathcal{J}}^{(K)}$ and forms two subproblems, or "branches:" one in which j is included in $\mathcal{J}^\star$ (i.e., added to $\underline{\mathcal{J}}$) and one in which it is excluded from $\mathcal{J}^\star$ (i.e., excluded from $\overline{\mathcal{J}}$).[15] The two fixed points resulting from applying the squeezing procedure to the bounding sets of each subproblem are the optimal decision sets *conditional* on the assumed inclusion or exclusion of j. The optimal decision set of the original CDCP is then the conditional optimal decision set that yields the higher value of π.

In cases where the fixed point of at least one of the subproblems does not contain two identical sets, the branching step can be applied *recursively*. In particular, within each subproblem we focus on another undetermined item j' and create two sub-sub-problems. Recursively applying the branching step in such a way creates a "tree," where the terminal nodes are subproblems for which the squeezing procedure has converged to a bounding set pair where the lower bound and upper bound are equal.

We now formally define the branching step.

Definition 6 (Branching step)**.** *Given bounding sets $[\underline{\mathcal{J}}, \overline{\mathcal{J}}]$, select some element $j \in \overline{\mathcal{J}} \setminus \underline{\mathcal{J}}$.*

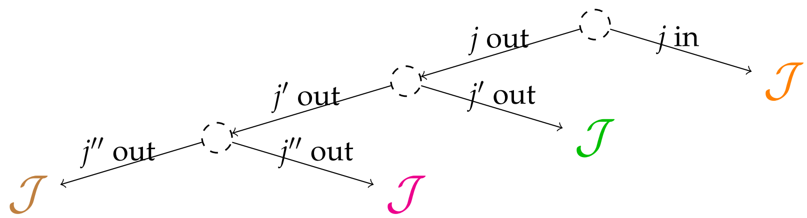
B$^{\mathbf{a}}$. *The mapping B^a is given by*

$$B^a([\underline{\mathcal{J}}, \overline{\mathcal{J}}]) \equiv \left\{ S^{a(K)}\left([\underline{\mathcal{J}} \cup \{j\}, \overline{\mathcal{J}}]\right), S^{a(K)}\left([\underline{\mathcal{J}}, \overline{\mathcal{J}} \setminus \{j\}]\right) \right\}$$

[14] For a given set of parameters, the structure of economic models typically implies that the return function exhibits the same type of SCD-C over the entire choice space. When out method is integrated into an estimation routine of the parameters determining the type of SCD-C, it is important to know ex-ante which type of SCD-C a given parameter guess induces in order to choose the appropriate squeezing step.

[15] Any item j such that $j \in \overline{\mathcal{J}}^{(K)} \setminus \underline{\mathcal{J}}^{(K)}$ can be chosen to initiate the branching procedure.

FIGURE 2: EXAMPLE PATH OF THE BRANCHING PROCEDURE



Notes: An example of a tree of subproblems resulting from applying the branching procedure recursively. Convergence on a single branch occurs when the squeezing procedure returns a conditionally optimal set, denoted by the colored $\mathcal{J}$s. The final output of the full recursive algorithm is the collection of all conditionally optimal sets.

B$^{\mathbf{b}}$. *The mapping* B^b *is given by*

$$B^b([\underline{\mathcal{J}}, \overline{\mathcal{J}}]) \equiv \left\{ S^{b(K)}\left([\underline{\mathcal{J}} \cup \{j\}, \overline{\mathcal{J}}]\right), S^{b(K)}\left([\underline{\mathcal{J}}, \overline{\mathcal{J}} \setminus \{j\}]\right) \right\}$$

For given initial bounding sets $[\underline{\mathcal{J}}, \overline{\mathcal{J}}]$, we denote the operator of applying the branching step until global convergence by $\mathrm{B}^{\mathrm{a}(K)}([\underline{\mathcal{J}}, \overline{\mathcal{J}}])$ and $\mathrm{B}^{\mathrm{b}(K)}([\underline{\mathcal{J}}, \overline{\mathcal{J}}])$, respectively. Global convergence of the branching step occurs when the stopping condition $\underline{\mathcal{J}} = \overline{\mathcal{J}}$ is met on each branch.[16]

Suppose the return function exhibits SCD-C from above.[17] Given an initial bounding pair $[\underline{\mathcal{J}}, \overline{\mathcal{J}}]$ with $\underline{\mathcal{J}} \subseteq \mathcal{J}^\star \subseteq \overline{\mathcal{J}}$, the globally converged result $\mathrm{B}^{\mathrm{a}(K)}([\underline{\mathcal{J}}, \overline{\mathcal{J}}])$ is a collection of branch-specific optimal decision sets. The cardinality of the set is the number of branches. Among these conditionally optimal decision sets, the one yielding the highest value of π is the optimal decision set solving the original CDCP. Note that contrary to the squeezing procedure, the branching procedure *always* identifies the optimal decision set.

For exposition, suppose the return function satisfies SCD-C, and consider Figure 2 which shows an example of a tree created by the branching procedure. It starts with a bounding set pair for which the squeezing procedure has converged, but there still remain undetermined items. One of these, j, is selected. Two branches based on this item are formed. The left hand branch corresponds to the subproblem where j is presumed to be excluded from the optimal decision set, while the right hand branch corresponds to the subproblem where j is presumed to be included. The squeezing procedure is reapplied in each branch. On the right hand branch, the squeezing procedure delivers

[16]Note that the definitions of the branching steps B^{a} and B^{b} suppose convergence of the squeezing procedure, so they are defined only when this convergence occurs. When then return function exhibits SCD-C, the squeezing procedure always converges.

[17]The same logic applies with $\mathrm{B}^{\mathrm{b}(K)}$ when the underlying return function exhibits SCD-C from below.

a bounding pair where $\underline{\mathcal{J}} = \overline{\mathcal{J}}$, yielding the orange $\mathcal{J}$. This decision set is optimal precisely conditional on the requirement that j must be included. On the other hand, convergence of the squeezing procedure in the left hand branch does not deliver an optimal decision set. The returned bounding pair is still such that there are strictly more items in the upper bound than lower bound set. The branching procedure therefore branches again, this time selecting the still undetermined item j' on which to branch. Repeating the squeezing procedure on both branches, the right hand branch once again delivers a conditionally optimal decision set, the green $\mathcal{J}$. This green decision set $\mathcal{J}$ is optimal conditional on both j and j' being included in the decision set. Again, the left hand branch does not deliver an optimal set, so the branching step is applied one last time, this time branching on item j''. This branch yields conditionally optimal decision sets, the brown and pink $\mathcal{J}$s. The first is optimal conditional on j, j', and j'' all being excluded. Likewise, the second is optimal conditional on excluding j and j', but including j''. As a final step, all conditionally optimal sets must be manually compared, by evaluating the return function with each. The decision set yielding the highest value is the global optimum.

To summarize the branching procedure, consider a CDCP as defined in equation 1 and let the bounding pair $[\underline{\mathcal{J}}, \overline{\mathcal{J}}]$ be such that $\underline{\mathcal{J}} \subseteq \mathcal{J}^\star \subseteq \overline{\mathcal{J}}$. Then, if π exhibits SCD-C from above,

$$\mathcal{J}^* = \underset{\mathcal{J} \in \mathsf{B}^{\mathsf{a}(K)}(\mathsf{S}^{\mathsf{a}}([\underline{\mathcal{J}}, \overline{\mathcal{J}}]))}{\arg\max} \pi(\mathcal{J})\,.$$

while if π exhibits SCD-C from below,

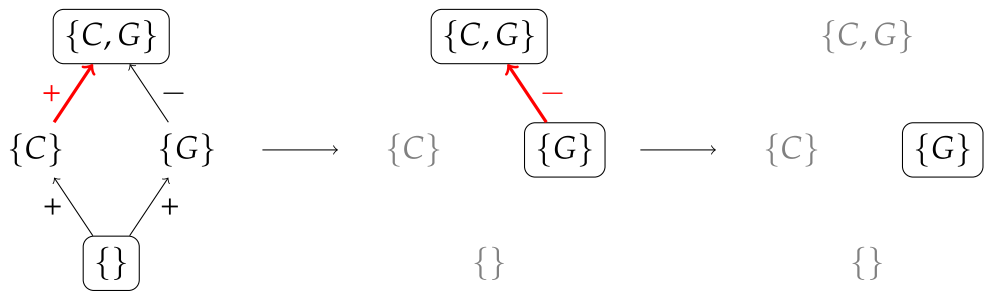
$$\mathcal{J}^* = \underset{\mathcal{J} \in \mathsf{B}^{\mathsf{b}(K)}(\mathsf{S}^{\mathsf{b}}([\underline{\mathcal{J}}, \overline{\mathcal{J}}]))}{\arg\max} \pi(\mathcal{J})\,.$$

A convenient property of the branching procedure is that it collapses to the brute force approach of manually comparing π across all elements in the choice space only in the worst-case scenario. If the squeezing procedure *never* updates the bounding sets, the branching procedure generates a tree of subproblems with as many terminal nodes as potential decision sets in the choice space, i.e., $2^{|J|}$. So as long as the squeezing procedure is effective at least on one branch, the branching procedure generally outperforms the brute force approach in convergence speed. In this sense, the branching procedure yields to the brute force method "one branch at a time."

3.4. Back to the Simple Example

Consider again the firm choosing optimal plant locations when its choice set is Germany and Canada. Suppose the firm's return function satisfies SCD-C from above.

FIGURE 3: A SIMPLE EXAMPLE: LOCATING PLANTS IN GERMANY AND CANADA



Notes: Applying the squeezing step three times to the simple Germany-Canada choice space. After each application, the choice space shrinks as countries are removed from the superset or added to the subset (denoted with a rectangle). Ultimately, one decision set remains.

Figure 3 depicts the steps of applying the squeezing method to the firm's plant location problem. The signs along the arrows indicate the marginal value of adding a given location to the bounding set. Consider the right most panel. The initial bounding pair is $[\{\}, \{G, C\}]$; $\mathcal{J}^\star \in [\underline{\mathcal{J}}, \overline{\mathcal{J}}]$, as required. Beginning with the subset, $\underline{\mathcal{J}} = \{\}$, we consider the return to adding Canada and Germany, separately. Since both countries have positive marginal values and SCD-C from above holds, we cannot discard either location as not optimal.

Next, we evaluate the respective marginal value of including Canada and Germany in the superset, $\overline{\mathcal{J}} = \{G, C\}$. Given SCD-C from above, Germany's marginal value remains positive when the Canadian plant is removed. The optimal decision set hence includes a German plant with certainty. We can draw no inference about Canada's inclusion in the optimal decision set. This completes the first application of the squeezing step.

The updated bounding sets in the middle panel of Figure 3 reflect that Germany is included in the optimal decision set with certainty. Since the marginal value of a Canadian plant is negative in this context, we conclude that the firm optimally only opens a plant in Germany, so that $\mathcal{J}^\star = \{G\}$.

4. Solving CDCPs for Heterogeneous Agents

In this section, we show how to solve for the policy function mapping agent type into optimal decision set in settings where a large number of heterogeneous agents each solve a CDCP. To that end, we introduce an additional restriction on the return function called single crossing differences in type, or SCD-T for short, that has implications on how the optimal decision set change with agent type.

4.1. Single Crossing Differences in Types

We begin by defining

$$\Lambda_j(\mathcal{J}) = \left\{\mathbf{z} \in \mathbf{Z} \mid D_j(\mathcal{J};\mathbf{z}) > 0\right\}$$

We define single crossing differences in type, or SCD-T, a restriction on the return function π.

Definition 7 (SCD-T)**.** *The return function π exhibits single crossing differences in type if, for all items j and decision sets $\mathcal{J}$, $\Lambda_j(\mathcal{J})$ and its complement $\Lambda_j(\mathcal{J})^c$ are both connected sets.*

The two contiguous sets $\Lambda_j(\mathcal{J})$ and $\Lambda_j(\mathcal{J})^c$ divide the typespace $\mathbf{Z}$ into types which receive positive marginal value and types which receive negative marginal value from j's inclusion in $\mathcal{J}$.[18]

Intuitively, the SCD-T restrictions implies that if the addition of item j to choice set $\mathcal{J}$ has positive marginal value for an agent of type $\mathbf{z}$, it also has a positive marginal value for an agent whose type is sufficiently close to $\mathbf{z}$ in the typespace.

If type heterogeneity is one-dimensional, we can write the SCD-T restriction in parallel with the SCD-C restriction in Section 3. In particular, given two types $z_1 < z_2$, SCD-T asserts, for all elements $j \in J$, decision sets $\mathcal{J}$, and aggregate vectors $\mathbf{y}$,[19]

$$D_j\pi(\mathcal{J};z_1,\mathbf{y}) \geq 0 \qquad\qquad \Rightarrow \qquad\qquad D_j\pi(\mathcal{J};z_2,\mathbf{y}) \geq 0\,.$$

When the marginal value function is strictly increasing in the type, z, the return function displays supermodularity between agent type and the decision set (and similarly submodularity when decreasing). We now provide a simple sufficient condition for SCD-T following from super- and sub-modularity between type and decision set. In particular, fix an item j, set $\mathcal{J}$, and component i of the multi-dimensional type vector. Holding all other components of the type vector fixed, one can check if the selected component i exhibits either supermodularity or submodularity with the decision set. If it does for every possible item j, set $\mathcal{J}$, and component i, then the return function exhibits SCD-T.[20]

[18]The SCD-T property is therefore implied by the supermodularity property introduced by Costinot (2009) (in its log form). Notice that our analysis is focused on partially ordered sets (lattices) not necessarily on totally ordered sets. We discuss below preliminary results related to the analysis in Costinot (2009) when the policy function obeys a *nesting* structure.

[19]Without loss of generality, we assume SCD-T holds from below. If it holds from above, then the problem can be recast in terms of $z' = 1/z$, which exhibits SCD-T from below.

[20]This sufficient condition still remains relatively general. For example, it allows for a given component i to be supermodular with item and decision set $(j,\mathcal{J})$, but submodular with different pair $(j',\mathcal{J})$. Likewise, it allows for a given component i to be supermodular with a pair $(j,\mathcal{J})$, while another component i' is submodular with the same pair.

In what follows, we use the SCD-T restriction to solve for the *policy function*, $\mathcal{J}^{\star}(\cdot)$, that maps agents' types to their optimal decision sets. We re-introduce the $\mathbf{z}$ indexing to indicate an agent's type, but continue to omit the aggregate state $\mathbf{y}$. Since the value of the return function depends on the agent's type, agents of different types may each have drastically different optimal decision sets. SCD-T restricts agents with similar types to have the same optimal decision set. For illustration, Figure 4 depicts a policy function associated with a single-dimensional typespace obeying these restrictions. In the figure, agents with types between z_1 and z_2 have the same optimal decision set $\mathcal{J}_1^{\star}$, while types between z_2 and z_3 instead optimally choose $\mathcal{J}_2^{\star}$, and so on. As a result of the SCD-T assumption, the corresponding policy function changes value only at interval boundaries, e.g., z_1 and z_2.

In the special case of single dimensional type heterogeneity with a the return function that obeys both SCD-C from below and SCD-T, the policy function obeys a *nesting* structure. That is, given two scalar types $z_1 < z_2$, it must be the case that $\mathcal{J}^{\star}(z_1) \subseteq \mathcal{J}^{\star}(z_2)$.[21] In the appendix, we generalize this nesting result to a multidimensional typespace under a stronger restriction in place of SCD-T. However, with SCD-C from *above* instead of *below* the policy function does not necessarily obey a nesting structure: there is no strict "hierarchy" of items, with the lowest type agents including only the first, then higher type agents further including the second, and so on. Instead, more productive agents may choose sets that contain less and different item than less productive agents, and vice versa. The resulting optimal policy function is a complicated object that is difficult to theoretically characterize. This challenge motivates our all-inclusive solution approach, which does not rely on any particular property of the policy function, and only requires SCD-C and SCD-T to hold for the underlying return function.

As illustrated in Figure 4, solving for the policy function requires *both* finding its "kink points," $z_1, z_2, \ldots$, and the optimal decision sets for the subregions of types they create. The "generalized squeezing procedure" we introduce next simultaneously solves for both.

4.2. The Generalized Squeezing Procedure

In the single agent CDCP in Section 3, we introduced the notion of bounding sets, $[\underline{\mathcal{J}}, \overline{\mathcal{J}}]$ associated with a CDCP. The set $\underline{\mathcal{J}}$ *includes* all items in $\mathcal{J}$ we know to *be* in the optimal decision set, while the

[21] Topkis (1978) shows that the policy function exhibits a nesting structure in settings with positive complementarities, a single dimension of agent heterogeneity, and supermodularity of agent type with choices, in which more productive types have optimal choice sets that nest those of less productive types (Antras et al. (2017) introduced this result to economics). We establish the nesting result in a setting with a multidimensional typespace in the case of positive spillovers, but also show that with negative spillovers no such results can be established. In fact, with SCD-C from below more productive types may find it optimal to choose strictly less items than less productive types.

FIGURE 4: THE POLICY FUNCTION OVER A ONE-DIMENSIONAL TYPESPACE

$$\underline{z} \overset{\mathcal{J}_1^\star}{\text{———————————}} z_1 \overset{\mathcal{J}_2^\star}{\text{—————}} z_2 \overset{\mathcal{J}_3^\star}{\text{————}} z_3 \overset{\mathcal{J}_4^\star}{\text{———————}} \overline{z}$$

Notes: The figure shows the one dimensional type space $[\underline{z}, \overline{z}]$ on a line. For illustration, it also shows groups of agent types that have the same optimal decision set. The single crossing in type assumption ensures that such groups exist. The resulting policy function $\mathcal{J}^\star(\cdot)$ changes its value only at each cutoff z_n, for $n = 1, 2, 3$.

set $\overline{\mathcal{J}}$ *excludes* all items in $\mathcal{J}$ we know to *not be* in the optimal decision set. As a result, $\underline{\mathcal{J}} \subseteq \mathcal{J}^\star \subseteq \overline{\mathcal{J}}$. With heterogeneous agents, we extend the notion of bounding sets to set-valued functions over the typespace, $\underline{\mathcal{J}}(\cdot)$ and $\overline{\mathcal{J}}(\cdot)$. These functions are such that $\underline{\mathcal{J}}(\mathbf{z}) \subseteq \mathcal{J}^\star(\mathbf{z}) \subseteq \overline{\mathcal{J}}(\mathbf{z})$ for any type $\mathbf{z}$.

With these concepts in hand, we introduce a "generalized squeezing step." The squeezing step in Section 3 acted on the two boundary sets associated with the CDCP. The generalized squeezing step instead acts on a region of the typespace Z such that for all $\mathbf{z} \in Z$ the current boundary set functions yield a constant value. We collect all information on a region Z relevant to the squeezing step in a 4-ple, $[(\underline{\mathcal{J}}, \overline{\mathcal{J}}, M), Z]$, where $\underline{\mathcal{J}}$ and $\overline{\mathcal{J}}$ are the values of the current bounding set functions over the interval Z. The set M is an "auxiliary" set which collects items the algorithm has already considered but could not make progress on.

Whereas an application of the squeezing step from Section 3 only updated the boundary sets, an application of the *generalized* squeezing step updates *both* the boundary sets and refines the partition of the typespace for which current boundary sets are identical (i.e., adds nee "kinks" to the boundary set functions). In particular, applying the generalized squeezing step to a given 4-ple creates up to three new 4-ples, each corresponding to a subregion of the original region Z, and each with either updated boundary sets or an updated auxiliary set. The technique is recursive, since the generalized squeezing step creates several 4-ples from an initial 4-ple at each application. The eventual output is a collection of 4-ples each with associated boundary sets. As with the simple squeezing procedure, ideally the boundary sets of each 4-ples coincide with one another in which case they also coincide with the optimal strategy for all types in the associated subregion of the typespace Z.

A natural initiation for applying the generalized squeezing procedure to a CDCP is to set the boundary sets to reflect that all items are undetermined, i.e., $\underline{\mathcal{J}}(\mathbf{z}) = \emptyset, \overline{\mathcal{J}}(\mathbf{z}) = J, \forall z \in Z$, and the auxiliary set to reflect that no item has been tried, i.e., $M = \emptyset$. Correspondingly, the initial 4-ple contains the entire typespace, i.e., $Z = \mathbf{Z}$.

We now define the "generalized squeezing step", which when applied to a 4-ple creates up to three new 4-tples each defined over a subregion of the typespace for which the original 4-ple was defined.

Definition 8 (Generalized squeezing step)**.** *Consider a CDCP faced by agents on a type space* $\mathbf{Z}$*, and a subregion of its typespace* $Z \subseteq \mathbf{Z}$ *with associated bounding sets* $(\underline{\mathcal{J}}, \overline{\mathcal{J}})$ *and auxiliary set* M*. Summarize it by the 4-ple* $[(\underline{\mathcal{J}}, \overline{\mathcal{J}}, M), Z]$ *and select some element* $j \in \overline{\mathcal{J}} \setminus (M \cup \underline{\mathcal{J}})$*.*

S^a. *The mapping* S^a *is defined as*

$$S^a([(\underline{\mathcal{J}}, \overline{\mathcal{J}}, M), Z]) \equiv \{[(\underline{\mathcal{J}} \cup \{j\}, \overline{\mathcal{J}}, \emptyset), \Lambda_j(\overline{\mathcal{J}})], [(\underline{\mathcal{J}}, \overline{\mathcal{J}} \setminus \{j\}, \emptyset), \Lambda_j(\underline{\mathcal{J}})^c)], \\ [(\overline{\mathcal{J}}, \underline{\mathcal{J}}, M \cup \{j\}), \Lambda_j(\underline{\mathcal{J}}) \setminus \Lambda_j(\overline{\mathcal{J}})]\} \ .$$

where any 4-ple with empty subregion may be omitted.

S^b. *The mapping* S^b *is defined as*

$$S^b([(\underline{\mathcal{J}}, \overline{\mathcal{J}}, M), Z]) \equiv \{[(\underline{\mathcal{J}} \cup \{j\}, \overline{\mathcal{J}}, \emptyset), \Lambda_j(\underline{\mathcal{J}})], [(\underline{\mathcal{J}}, \overline{\mathcal{J}} \setminus \{j\}, \emptyset), \Lambda_j(\overline{\mathcal{J}})^c)], \\ [(\overline{\mathcal{J}}, \underline{\mathcal{J}}, M \cup \{j\}), \Lambda_j(\overline{\mathcal{J}}) \setminus \Lambda_j(\underline{\mathcal{J}})]\} \ .$$

where any 4-ple with empty subregion may be omitted.

We use an example, to show how to use the generalized squeezing step. Consider a CDCP with a single dimension of heterogeneity and with a return function that satisfies SCD-T and SCD-C from above. In the initial 4-ple Z is set to the entire typespace $\mathbf{Z}$, $\underline{\mathcal{J}}(\cdot) = \emptyset, \overline{\mathcal{J}}(\cdot) = J, \forall z \in Z$, and $M = \emptyset$. To apply the generalized squeezing step S^a to the corresponding 4-ple $[(\underline{\mathcal{J}}, \overline{\mathcal{J}}, M), Z]$, we first choose an undetermined item $j \in \overline{\mathcal{J}} \setminus \underline{\mathcal{J}}$. We can identify the "cutoff" agent types $z^{\text{in}}, z^{\text{out}} \in Z$ which are exactly indifferent between including j in $\overline{\mathcal{J}}$ and $\underline{\mathcal{J}}$ respectively:[22]

$$0 = D_j\pi(\overline{\mathcal{J}}; z^{\text{in}}) \qquad\qquad 0 = D_j\pi(\underline{\mathcal{J}}; z^{\text{out}}) \ .$$

These cutoff types divide the original region Z into *up to* three subregions.

For all types $z \in Z$ with $z < z^{\text{out}}$,

$$0 = D_j\pi(\underline{\mathcal{J}}; z^{\text{out}}) \qquad \Rightarrow \qquad 0 \geq D_j\pi(\mathcal{J}^\star(z); z^{\text{out}}) \qquad \Rightarrow \qquad 0 \geq D_j\pi(\mathcal{J}^\star(z); z) \ .$$

The first inequality follows from SCD-C, since $\underline{\mathcal{J}} \subseteq \mathcal{J}^\star(z)$ for all $z \in Z$. The second inequality follows from SCD-T. We can then conclude that all types $z \in Z$ below z^{out} exclude j from their

[22] It is irrelevant whether the type mass function $f(\cdot)$ assigns positive values to the cutoff values $z^{\text{in}}, z^{\text{out}} \in Z$.

optimal decision set. Likewise, for all $z \in Z$ with $z > z^{\text{in}}$,

$$0 = D_j \pi(\overline{\mathcal{J}}; z^{\text{in}}) \qquad \Rightarrow \qquad 0 \leq D_j \pi(\mathcal{J}^\star(z); z^{\text{in}}) \qquad \Rightarrow \qquad 0 \leq D_j \pi(\mathcal{J}^\star(z); z)$$

Again, the first inequality follows from SCD-C, since $\mathcal{J}^\star(z) \subseteq \overline{\mathcal{J}}$ for all $z \in Z$, and the second from SCD-T. Given SCD-T and SCD-C, it is easy to verify that $z^{\text{out}} \leq z^{\text{in}}$.

The two cutoffs we identified create three new 4-ples. After dividing Z into three new subregions according to the two cutoffs z^{in} and z^{out}, we update each subregion's bounding and auxiliary sets with the new information. For all types in the right subregion, j is optimally included, so the subset $\underline{\mathcal{J}}$ includes j. For all types in the left subregion, j is optimally excluded, so the superset $\overline{\mathcal{J}}$ excludes j. For all types in the middle subregion, we cannot conclude that j is either optimally included or excluded. Its bounding sets, $\underline{\mathcal{J}}$ and $\overline{\mathcal{J}}$, are the same as those of its "parent" region Z. Instead we add j to M to encode the information that, given the current bounding sets $[\underline{\mathcal{J}}, \overline{\mathcal{J}}]$ on Z, j remains undetermined. Consequently, the set M must be "reset" to the empty set whenever either bounding set, $\underline{\mathcal{J}}$ or $\overline{\mathcal{J}}$, is non-trivially updated since this information changes the potential marginal value of the items in M.[23] We have now created three new subregions from the original subregion Z, each with at least one of the original $\underline{\mathcal{J}}$, $\overline{\mathcal{J}}$, or M updated.[24] Note that it could be that $z^{\text{out}} \notin Z$ of $z^{\text{in}} \notin Z$ or $z^{\text{out}}, z^{\text{in}} \notin Z$ in which case the generalized squeezing step creates less than three new 4-ples.

The next theorem establishes that if a CDCP's underlying return function exhibits SCD-C and SCD-T, then each application of the generalized squeezing step updates the bounding sets without excluding items that are part of the optimal decision for any subregion Z of the typespace.

Theorem 2. *Consider a CDCP as defined in equation (1) for agents on a typespace* $\mathbf{z}$*, and associated 4-ple* $[(\underline{\mathcal{J}}_0, \overline{\mathcal{J}}_0, M), Z]$ *for which* $M \subseteq (\overline{\mathcal{J}}_0 \setminus \underline{\mathcal{J}}_0)$ *and* $\overline{\mathcal{J}}_0 \subseteq \mathcal{J}^\star(\mathbf{z}) \subseteq \underline{\mathcal{J}}_0$ *for all* $\mathbf{z} \in Z$*. Suppose the underlying return function exhibits SCD-T over* Z*.*

1. *If the underlying return function* π *exhibits SCD-C from above, then applying* S^a *recursively partitions* Z *into disjoint subregions. Further,* $\underline{\mathcal{J}}_0 \subseteq \underline{\mathcal{J}}(\mathbf{z}) \subseteq \mathcal{J}^\star(\mathbf{z}) \subseteq \overline{\mathcal{J}}(\mathbf{z}) \subseteq \overline{\mathcal{J}}_0$ *for each* $\mathbf{z} \in Z$*.*
2. *If the underlying return function* π *exhibits SCD-C from below, then applying* S^b *recursively partitions* Z *into disjoint subregions. Further,* $\underline{\mathcal{J}}_0 \subseteq \underline{\mathcal{J}}(\mathbf{z}) \subseteq \mathcal{J}^\star(\mathbf{z}) \subseteq \overline{\mathcal{J}}(\mathbf{z}) \subseteq \overline{\mathcal{J}}_0$ *for each* $\mathbf{z} \in Z$*.*
3. *Conditional on the appropriate SCD-C restriction, each the recursive application of* S^a *and* S^b *converges*

[23] Observe that the item j must therefore be chosen from $\overline{\mathcal{J}} \setminus (M \cup \underline{\mathcal{J}})$. The reason is that, for the current bounding sets, all items in M have already been considered with no progress made.

[24] Returning to the definition of S^a, we can verify the above example in single dimensional typespace precisely corresponds to one application of the generalized squeezing step. As a technical detail, it is possible for either z^{out} or z^{in} to be outside of Z. In these cases, the generalized squeezing step will return one or two subintervals instead of three. The formal definitions of the generalized squeezing steps allow for this possibility.

in $O(n)$ time.

Given a CDCP with underlying return function exhibiting SCD-C and SCD-T, we can define the generalized squeezing procedure as recursively applying the generalized squeezing step until $\overline{\mathcal{J}} = M \cup \underline{\mathcal{J}}$ on each subregion of the typespace. The lower portion of Figure **??** visualizes a possible "tree" of subregions that emerges from applying the generalized squeezing step repeatedly to each new 4-ple.

Once $\overline{\mathcal{J}} = M \cup \underline{\mathcal{J}}$ for each subregion, there remain no undetermined elements j on which to make progress. The squeezing procedure has converged globally when it converges on all 4-ples separately. We denote the operator that encodes the recursive application of the generalized squeezing step until global convergence by $\mathrm{S}^{\mathrm{a}(K)}$ and $\mathrm{S}^{\mathrm{b}(K)}$. Given Theorem 2, the generalized squeezing procedure delivers bounding set functions $\underline{\mathcal{J}}(\cdot)$ and $\overline{\mathcal{J}}(\cdot)$ such that $\underline{\mathcal{J}}(\mathbf{z}) \subseteq \mathcal{J}^\star(\mathbf{z}) \subseteq \overline{\mathcal{J}}(\mathbf{z})$ for all $\mathbf{z} \in \mathbf{Z}$.

For subregions where $\underline{\mathcal{J}}(\mathbf{z}) \subset \overline{\mathcal{J}}(\mathbf{z})$ for some type $\mathbf{z}$ after global convergence of the generalized squeezing procedure, we provide a generalized branching procedure in the next section.

4.3. Generalized Branching Procedure

We define the generalized branching step as follows.

Definition 9 (Generalized branching step)**.** *Given a 4-ple $[(\underline{\mathcal{J}}, \overline{\mathcal{J}}, M), Z]$ and some $j \in M$,*

B$^{\mathbf{a}}$. *The mapping B^a is given by*

$$B^a([(\underline{\mathcal{J}}, \overline{\mathcal{J}}, M), Z]) \equiv S^{a(K)}([\underline{\mathcal{J}} \cup \{j\}, \overline{\mathcal{J}}, \emptyset, Z]) \cup S^{a(K)}([\underline{\mathcal{J}}, \overline{\mathcal{J}} \setminus \{j\}, \emptyset, Z])\,.$$

where $S^{a(K)}$ denotes recursively applying S^a until convergence.

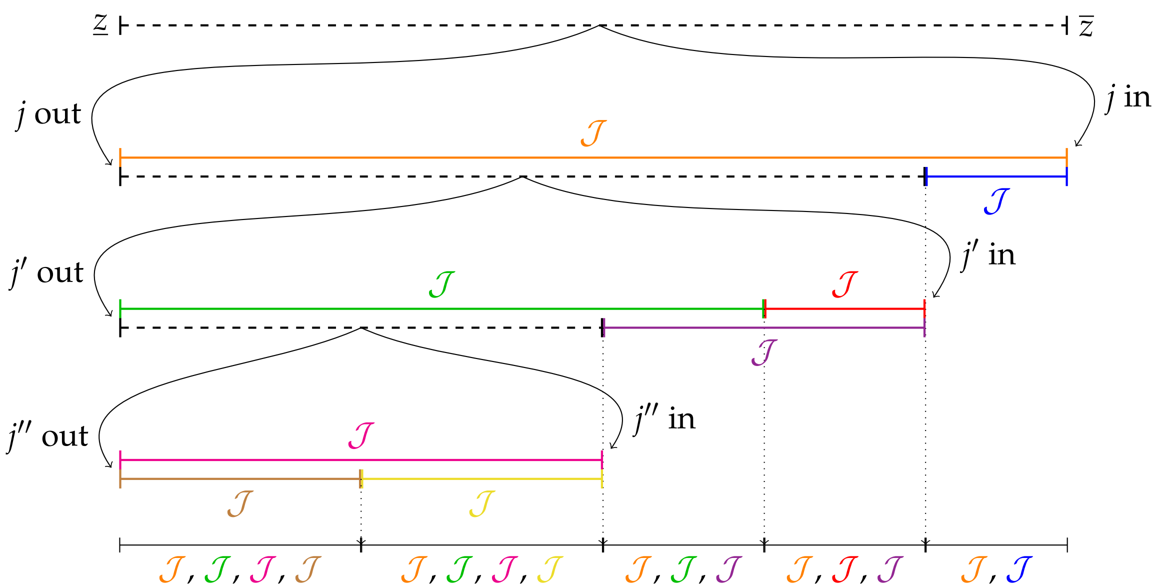
B$^{\mathbf{b}}$. *The mapping B^b is given by*

$$B^b([(\underline{\mathcal{J}}, \overline{\mathcal{J}}, M), Z]) \equiv S^{b(K)}([\underline{\mathcal{J}} \cup \{j\}, \overline{\mathcal{J}}, \emptyset, Z]) \cup S^{b(K)}([\underline{\mathcal{J}}, \overline{\mathcal{J}} \setminus \{j\}, \emptyset, Z])\,.$$

where $S^{b(K)}$ denotes recursively applying S^b until convergence.

Given an initial 4-ple $[(\underline{\mathcal{J}}, \overline{\mathcal{J}}, M), Z]$ and an undetermined item $j \in M$, the generalized branching step creates two branches or subproblems. The first supposes that j is included in the optimal decision set, while the second supposes that it is excluded. To each of these two subproblems we

FIGURE 5: THE GENERALIZED BRANCHING PROCEDURE: AN EXAMPLE OUTCOME



Notes: At each application, one undetermined item is selected on which to branch, yielding conditional policy functions. Each coloured $\mathcal{J}$ represent a different decision set. Branching continues until no undetermined items remain for all types on all branches. Conditionally optimal decision sets for each type are ultimately gathered at the bottom of the figure.

apply the generalized squeezing procedure until global convergence obtaining a collection of 4-ples which exhaustively partition the original region Z.[25] Each branch may now contain different partitions of the original typespace.

On either branch, if there are any 4-ples with undetermined items, the generalized branching step can be applied again. The generalized branching procedure consists in recursively applying the generalized branching step this way, where recursion stops on a given 4-ple when the bounding sets for that 4-ple coincide. Global convergence occurs when bounding sets coincide for 4-ples on all branches. Then, the output of the generalized branching procedure is a collection of 4-ples each of the form $[\mathcal{J}, \mathcal{J}, \emptyset, Z]$.

For illustration, Figure 5 depicts the process of applying the generalized branching procedure to an initial 4-ple $[(\underline{\mathcal{J}}, \overline{\mathcal{J}}, M), Z]$. The initial 4-ple specifies lower and upper bound sets $(\underline{\mathcal{J}}, \overline{\mathcal{J}})$ over the entire dotted interval Z between $\underline{z}$ and $\overline{z}$. Applying the generalized branching step once, the problem is divided into two subproblems: one corresponding to requiring item j to be excluded, and the other requiring that item j be included. In the subproblem on the right branch, the squeezing

[25]The definitions of the branching steps B^a and B^b suppose convergence of the generalized squeezing procedure, and are therefore defined only when convergence occurs.

procedure identifies the single (orange) decision set that is optimal for the whole interval conditional on including j. On the left branch, j is excluded. In this case, convergence from the squeezing procedure delivers a policy function only for the highest types $z \in Z$, identifying the (blue) optimal decision set. Undetermined elements remain for the lower types of the range. The branching step is thus reapplied for this subsection of the original interval, selecting a second undetermined element, j'. This procedure repeats until no undetermined elements remain in any of the branches for any type $z \in Z$.

Now consider the entire initial region Z, repeated at the bottom of the graphic. The repeated application of the squeezing procedure to smaller and smaller subregions of the typespace creates subregions of the overall typespace that share several conditional optimal policy functions. We show the conditional optimal policy function that apply to each subregion. For each subregion, we now manually choose which of the associated conditional policy functions maximizes the return function for each type in the subregion. Piecing together the so chosen optimal policy functions for each interval yields the optimal policy function that solves the original CDCP on the interval $[\underline{z}, \overline{z}]$.

4.4. Back to the Simple Example

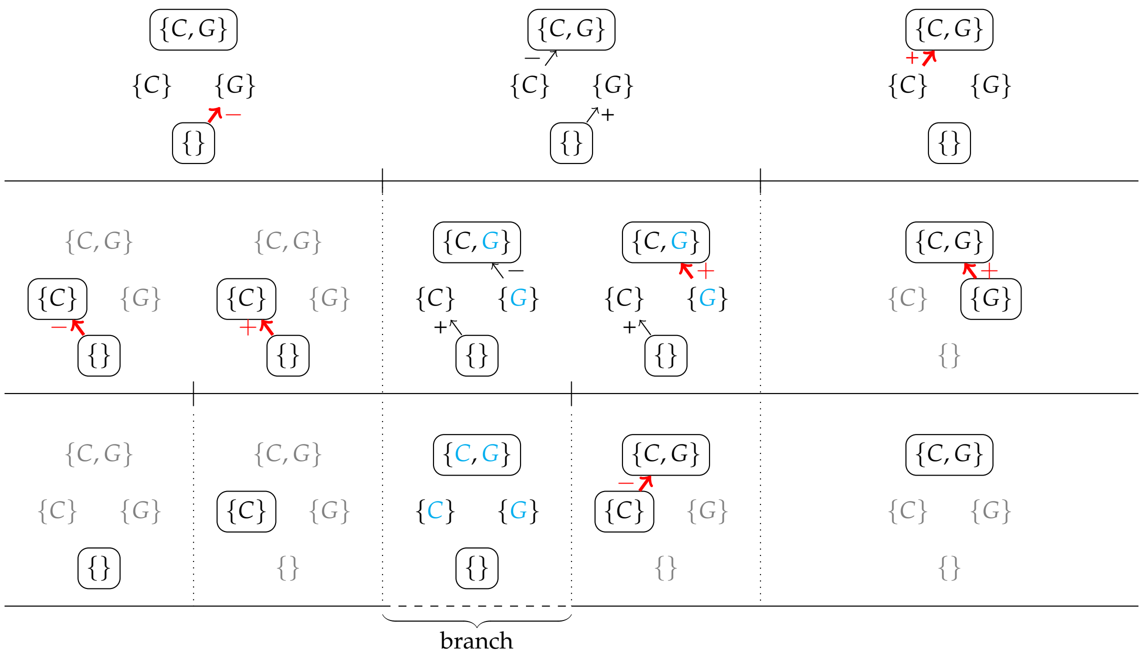
We now consider the simple example from before, but with several firms whose return functions satisfies SCD-C from above and SCD-T. Each row of Figure 6 illustrates one application of the generalized squeezing step. In the first row, the entire interval $\mathbf{Z}$ shares an identical bounding pair $[\{\}, \{C, G\}]$ indicated with rectangles. As before, the signs along the arrows indicate the marginal value of adding a given location to the bounding set. Red arrows imply that an interval's bounding pair can be updated.

In the first row, we consider adding a German plant. We identify two cutoff productivities, marked with vertical ticks. For firm types below the first cutoff, adding a plant in Germany when there is no plant in Canada yields negative marginal benefit; these firms never open a plant in Germany. For firm types above the second cutoff, adding a plant in Germany while there is no plant in Canada yields positive marginal benefit; these firms always open a plant in Germany. For firms in the middle interval, no update is possible.

The middle row of Figure 6 shows the typespace with updated bounding pairs. For the middle interval, G enters the auxiliary set M since no decision could be reached yet; we encode this update by representing Germany in blue.

In the leftmost interval, we identify the cutoff productivity for which opening a plant in Canada has positive marginal value. In the rightmost interval, all types receive positive marginal value from

FIGURE 6: AN EXAMPLE OF THE GENERALIZED SQUEEZING PROCEDURE



Notes: A possible recursive sequence from the generalized squeezing procedure.

opening a German plant. In the middle interval, we identify a cutoff above which firms receive positive marginal benefit from a Canadian plant when a German plant exists.

The third row of Figure 6 reflects updates from the second line. We have found the optimal decision set for intervals one, two, and five. Interval four has to be updated one more time, to reflect that these types optimally open a Canadian plant only. For interval three, both C and G are in the auxiliary set and there remain no countries to consider; the branching step has to be applied (not shown).

5. Plant Location and Input Sourcing Problems as CDCPs

In this section, we provide generalizations two canonical CDCPs – the plant location and the input sourcing problems – and show the conditions under which they satisfy SCD-C and SCD-T.

5.1. Plant Location Problems

We present a generalized plant location problem that nests several models studied in the literature and provide sufficient conditions for SCD-C and SCD-T to hold. We conclude with a numerical exercise to illustrate the computational and accuracy gains from applying our solution method.

Framework The economy consists of a discrete set $j = 1, \ldots, J$ of locations. Firms choose a set of locations $\mathcal{J} \subseteq J$ in which to open a plant to produce their firm-specific good. Opening a plant in location j occurs a j-specific fixed costs f_j. Firms can use a plant in location j to supply their good to other locations k.

Firms differ in their type (or headquarter productivity) and their plant productivity. Both characteristics may affect its revenues in market k. The firm knows its type $\mathbf{z}$ when making its plant location decision. The probability density of firm types is denoted $f(\mathbf{z})$ and can be degenerate. The plant productivity, $\boldsymbol{\varphi}_j$, collects the characteristics unknown to the firm when choosing its plant locations, but known when choosing the optimal plant j to serve market k. The probability density for plant productivity is denoted G_j and could be degenerate. The firm maximizes its *expected return* when choosing its plant locations $\mathcal{J}$ prior to the realization of $_j$.

We denote the potential returns of serving market k from a plant in location j by $r_{jk}(\boldsymbol{\varphi}_j; \mathbf{z}, \mathbf{y})$. The term $r_{jk}(\boldsymbol{\varphi}_j; \mathbf{z}, \mathbf{y})$ can vary by location pair, allowing for bilateral trade cost, location-specific costs, and market-specific (endogenous or exogenous) conditions. We assume that goods markets k are separable: the firm chooses the optimal plant $j \in J$ to serve market k by solving $\max_j \left[r_{jk}(\boldsymbol{\varphi}_j; \mathbf{z}) \right]$, for each k separately.

The firm's overall return function can be written

$$\pi(\mathcal{J}; \mathbf{z}, \mathbf{y}) = \sum_{k \in K} \mathbb{E} \left[\max_{j \in \mathcal{J}} r_{jk}(\boldsymbol{\varphi}_j; \mathbf{z}, \mathbf{y}) \,\middle|\, \mathbf{z}, \mathbf{y} \right] - \sum_{j \in \mathcal{J}} f_j, \tag{3}$$

where the fixed costs of opening plant in all locations contained in $\mathcal{J}$ are deducted from operating profits.[26] We show below that this formulation encompasses many variations of the plant location (and input sourcing) problem studied in the literature.

Single Crossing Differences Conditions We now present a set of sufficient conditions under which SCD-C and SCD-T hold in this environment.

[26] Note that the fixed costs can be firm-specific in which case they would form part of the (multidimensional) firm type $\mathbf{z}$.

Suppose that each $\boldsymbol{\varphi}_j$ is drawn independently from each other $\boldsymbol{\varphi}_{j'}$.[27] As a result, we can express the expectation term in equation 3 as follows:

$$\bar{r}_k(\mathcal{J};\mathbf{z},\mathbf{y}) \equiv \mathbb{E}\left[\max_{j\in\mathcal{J}} r_{jk}(\boldsymbol{\varphi}_j;\mathbf{z},\mathbf{y}) \,\middle|\, \mathbf{z},\mathbf{y}\right] = -\underline{r} + \int \left[1 - \prod_{j\in\mathcal{J}} \tilde{G}_{jk}(r;\mathbf{z},\mathbf{y})\right] \mathrm{d}r$$

where $\underline{r}$ is the lower bound on the value of $r_{jk}(\cdot;\mathbf{z},\mathbf{y})$ and $\tilde{G}_{jk}(\cdot;\mathbf{z},\mathbf{y})$ is the distribution of plant potential revenues conditional on firm type and aggregate state. The distribution $\tilde{G}_{jk}(\cdot;\mathbf{z},\mathbf{y})$ is a transformation of the distribution of plant productivity, $G_j(\cdot)$.

The more locations are in $\mathcal{J}$ the smaller the product on the right, and the larger the value of $\bar{r}_k(\mathcal{J};\mathbf{z})$. Since the firm chooses the most profitable among its plants æ to serve market k the expected return from market k must weakly increase when another plant is added to J. However, the marginal value of a plant in location j declines in the total number of locations $\mathcal{J}$: the additional location j only adds positive value when its chosen to serve market k, which becomes (weakly) less likely the more plants the firm operates. As a result, the function $\bar{r}_j(\cdot;\mathbf{z},\mathbf{y})$ exhibits SCD-C from above if the φ_js are distributed independently from each other.

The following proposition establishes that i.i.d. draws of φ_j are also sufficient for SCD-C from above of the overall return function, π.

Proposition 1 (Conditions for SCD-C)**.** *Fixing any* $\mathbf{z}$, *suppose the* $\boldsymbol{\varphi}_j$ *vectors are independent across locations* j. *Then, the profit function* π *satisfies SCD-C from above.*

This sufficient condition formalizes the intuition discussed above – that each operating location j serves as a (partial) substitute to each other location, so that the benefit of any additional location falls as it is added to a growing set. Note that, besides independence, no distributional assumptions need be imposed on the $\boldsymbol{\varphi}_j$ draws, nor is any other assumption needed on the primitives of the model.

Establishing SCD-T requires more structure on the nature of firm heterogeneity. We provide a number of sufficient conditions for SCD-T, each applying to a different formulation of firm heterogeneity. Note that the distribution of firm heterogeneity is irrelevant for SCD-C or SCD-T.

Proposition 2 (Sufficient conditions for SCD-T)**.** *For each* $\mathbf{z}$, *suppose the* $\boldsymbol{\varphi}_j$ *vectors are independent across locations* j. *Then, each of the following are sufficient conditions for the return function to exhibit SCD-T.*

a) Suppose that $\boldsymbol{\varphi}_j$ *are distributed independently of* $\mathbf{z}$ *and the functions* $r_{jk}(\cdot;\cdot)$ *are multiplicatively or*

[27] Note we allow the components within each vector to have arbitrary correlation with each other.

additively separable; that is $r_{jk}(\cdot;\cdot)$ can be written

$$r_{jk}(\boldsymbol{\varphi}_j;\mathbf{z}) = \alpha(\mathbf{z})\beta_{jk}(\boldsymbol{\varphi}_j) \qquad \text{or} \qquad r_{jk}(\boldsymbol{\varphi}_j;\mathbf{z}) = \alpha(\mathbf{z}) + \beta_{jk}(\boldsymbol{\varphi}_j)\,.$$

Then, the return function satisfies SCD-T.

b) *Suppose that $\boldsymbol{\varphi}_j$ are distributed independently of $\mathbf{z}$ and that the firm characteristics can be partitioned into vectors $\mathbf{z}_j$ so that for all $\boldsymbol{\varphi}_j$, $r_{jk}(\boldsymbol{\varphi}_j;\cdot)$ depends only on $\mathbf{z}_j$ and not on $\mathbf{z}_{j'}$ for $j' \neq j$. If, given any $\boldsymbol{\varphi}_j$, each partial derivative*

$$\frac{\partial}{\partial z_{j,i}} r_{jk}(\boldsymbol{\varphi}_j;\mathbf{z}_j) \qquad\qquad \mathbf{z}_j = \{z_{j,1},\ldots,z_{j,i},\ldots\}$$

never changes sign over the typespace, the return function satisfies SCD-T.

c) *Suppose the distributions of $\boldsymbol{\varphi}_j$ depend on $\mathbf{z}$ in such a way that there is a change of variables $\boldsymbol{\eta}_j$ where $\boldsymbol{\eta}_j$ is independent of $\mathbf{z}$. Define*

$$\tilde{h}_{jk}(\boldsymbol{\eta}_j;\mathbf{z}) = r_{jk}(\boldsymbol{\varphi}_j;\mathbf{z})\,.$$

If $\tilde{h}_{jk}$ satisfies conditions a) or b), then the return function exhibits SCD-T.

The first condition applies to potential return functions r_{jk} that are either additively or multiplicatively separable. The second condition restricts instead how potential returns r_{jk} vary with firm type. The final sufficient condition is applicable to settings where the distribution of location-specific draws $\boldsymbol{\varphi}_j$ are themselves parameterized by firm characteristics $\mathbf{z}$.

Plant Location Problems in the Literature In this section, we introduce some of the models in the literature that map into our general framework. Table 1 shows the potential return function of serving market k from location j in a set of different papers. The unifying feature of models with or without uncertainty about plant productivity is the presence of the max operator in equation 3 which makes individual plants substitutes.

Tintelnot (2017) studies a multinational plant location problem with one-dimensional firm heterogeneity. In this framework, the r_{jk} functions are the variable profit under CES preferences, where the expectation is taken over idiosyncratic draws of plant productivities. Since plant productivities are drawn independently from each other, the problem satisfies SCD-C from above. Since productivity is multiplicatively separable under the CES monopolistic competition formulation, Proposition 2c guarantees that SCD-T holds.

TABLE 1: SPECIAL CASES OF THE BENCHMARK MODEL IN THE LITERATURE

Problem Formulation	r_{jk}	SCD-T
Tintelnot (2017)	$r_{jk}(\varphi_j; z, \mathbf{y}) = \frac{1}{\sigma} X_k \left(\frac{\sigma}{\sigma-1} \frac{w_j d_{jk}}{\varphi_j P_k} \right)^{1-\sigma}$	2c
Arkolakis et al. (2018)	$r_{jk}(\mathbf{z}, \mathbf{y}) = \frac{1}{\sigma} X_k \left(\frac{\sigma}{\sigma-1} \frac{d_{jk} w_j}{z_j P_k} \right)^{1-\sigma}$	2b
SPLP (e.g., Balinski (1965))	$r_{jk} = -c_{jk}$	N/A

Notes: This Table shows how frameworks from the literature map into our generalized framework.

Arkolakis et al. (2018) features a model of global innovation, where firms choose plant locations while heterogeneous over a vector of independent location-specific productivities. The r_{jk} functions are similar to those in Tintelnot (2017) and the problem similarly satisfies SCD-C from above. Given the multidimensional firm typespace, the problem satisfies SCD-T as an implication of Proposition 2b.

In operations research the so-called simple plant location problem (SPLP) describes an important family of discrete, deterministic, NP-hard, and widely applicable optimization problems (see, e.g., Balinski (1965) or Owen and Daskin (1998)). In the standard formulation there is a set of locations, each with a constant demand for the firm's product. The firm needs to serve all demand points, and minimizes the cost of doing so. There is no uncertainty over plant productivity, so that G_{ij} is degenerate. Serving location k from j incurs a cost of $c_{jk} > 0$. The firm chooses its set of plant locations $\mathcal{J}$ and then chooses for each k the cost-minimizing supplier. The cost minmization problem is the dual of the profit maximization problem, and so the max operator again implies that individual plants are (weak) substitutes. As a result, the problem satisfies SCD-C from above. The SPLP is concerned with a single plant only, so that SCD-T is not relevant.

Plant Location Applications In this section, we apply our solution methods to the model Tintelnot (2017). We solve for the policy function mapping firm type into optimal decision set. We contrast three different techniques to solve for the policy function: (1) discretize the firm type space and solve the CDCP of the firm at each grid point using brute force, (2) discretize the firm type space and solve the CDCP of the firm at each grid point using the squeezing method outlined in Section 3 above, and (3) using the generalized squeezing method from Section 4.

Table 1 above showed how the model in Tintelnot (2017) maps into our general framework. Different from before, we now consider firms in several countries and index each firm's return function by the country i in which it is headquartered. A firm's sales in market k depend on total local consumption expenditure, X_k, the local price index, P_k, and the efficiency of its plant network summarized in the term Θ_{ik}. We denote by γ_{ij} the communication cost between the headquarter in location i and a plant in location j, and the trade cost between a plant in location j and destination market k by d_{jk}. The higher the communication and trade costs, the lower the efficiency of the firm's production network. Firm types are single-dimensional and their probability density, $f_i(\cdot)$, is Pareto with shape parameter ξ and minimum value $z_{\min}$. The distribution of plant productivity (φ_{ij}), $G_{ik}(\cdot)$, is Fréchet with shape parameter θ and with the firm type, z, as the shape parameter for each firm. As a result, average plant productivity is higher for firms with higher firm type. [28]

In summary, we can write the firm's profit function in equation 3 as follows:

$$\pi_i(\mathcal{J};z,\mathbf{y}) = \tilde{\Gamma} z \sum_k \frac{1}{\sigma} X_k \left(\frac{\sigma}{\sigma-1}\frac{1}{P_k}\right)^{\frac{1}{1-\sigma}} \Theta_{ik}\left(\mathcal{J}\right)^{\frac{\sigma-1}{\theta}} - \sum_{j\in\mathcal{J}} w_j f_j \tag{4}$$

where $\Theta_{ik} = \sum_{j\in\mathcal{J}}\left(w_j\gamma_{ij}d_{jk}\right)^{-\theta}$, σ is the elasticity of substitution of varieties in consumer utility. Relative to equation 3, equation 4 incorporates the fact that the maximization problem has a closed form solution given the Fréchet assumption on the the plant productivity distribution. The aggregate state, $\mathbf{y} = \{X_i, w_i, P_i\}_i$, contains the vectors of all general equilibrium objects (wages, total expenditure, and price indices) for each country. As discussed above, this profit function satisfies both SCD-C from above and SCD-T of type $2c$.

We choose the following parameters for the model. Following Tintelnot (2017), we set the shape parameter of the plant productivity Fréchet distribution $\theta = 7$ and consumer elasticity of substitution $\sigma = 6$. We assume productivities are drawn from a Pareto distribution with minimum and shape specific to the origin country.

The other parameters more particular to the context to which it is calibrated. To show the performance of our techniques in various contexts, we solve the model a random draw of the remaining parameters (f_j, f_j^e, L_j, T_j, $z_i^{\min}$, ξ_i, d_{jk}, γ_{ij}). We draw the fixed costs of establishing a plant f_j each from a uniform random distribution over the values $(3,4)$. We similarly draw the fixed costs of entry f_j^e, the labor forces L_j, the Fréchet scales T_j, the Pareto minimums $z_i^{\min}$, and the Pareto shapes ξ_i each from a uniform random distribution over the values $(1,2)$. Finally, we set $\gamma_{ii} = 1$ and draw $\gamma_{ij}\,\forall i \neq j$ from a uniform random distribution over the values $(1,2)$. We do the same for bilateral trade costs d_{jk}.

[28] Note that the distribution of firm types is irrelevant for the individual firm's problem and for our method of finding the policy function, conditional on the aggregate state $\mathbf{y}$. The distribution of firm type is only needed when used in conjunction with the the policy function to compute the model's general equilibrium aggregates which are stored in $\mathbf{y}$. In other words, conditional on yCDCP is independent of firm distribution.

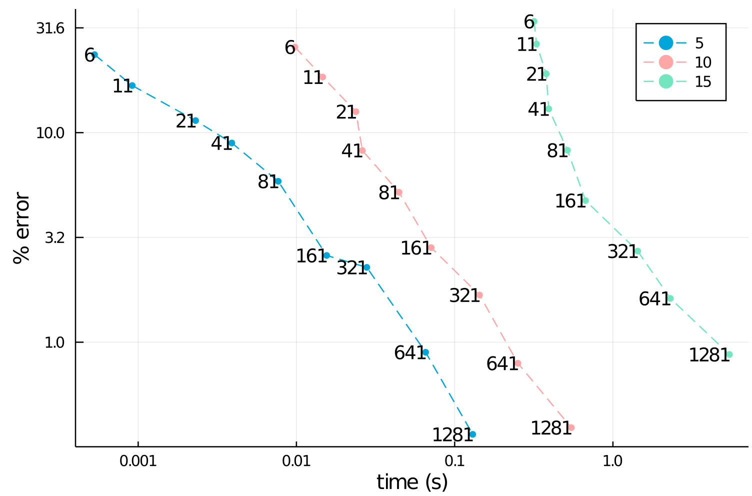
To problem of each firm depends on the aggregate state **y** which the firm takes as given. We choose **y** as follows: for each trial run, we first compute the equilibrium of the model in the absence of multinational production (MP), where each firm only has a plant in the location of its headquarter which yields $\mathbf{y}^{\text{No-MP}}$. Then we solve for the policy function conditional on the aggregate state $\mathbf{y}^{\text{No-MP}}$.

Table 7 shows the time of solving for the policy function of firms in each country, separately for three different methods. To generate the numbers in Columns (1) and (2), we discretize the firm typespace into 81 points to approximate the policy function. For each type on the grid, Column (1) solves the CDCP using the brute force method of evaluating all possible decision sets and then choosing the profit-maximizing one. Column (2) instead uses the single-agent method outlined in Section 3 for each type on the grid. Column (3) does not discretize the typespace, but instead solves for the exact policy function using the method outlined in Section 4 above. The results reveal that with larger numbers of countries, the single-agent method we outline is about three orders of magnitudes faster on average than brute force. Using the explicit aggregation method we introduce reduced the time to compute the policy function by another order of magnitude or two with a combined computational gain of about five order of magnitude.

The method described in Section 4 besides generally being the fastest method, has the added advantage of being exact. The policy functions computed using the techniques in Columns 1 and 2 are approximations only, necessitating interpolation for productivity values in between gridpoints. With more gridpoints (a denser grid), interpolation error necessarily decreases at the cost of computation time. The graph in Figure 7 explores this tradeoff for 5 countries, 10 countries, and 15 countries. For each point in the graph, we approximate the policy function using the single-agent method on a discretized grid, using the number of gridpoints indicated. For example, the point in the top left of the graph represents the outcome with a discretization of 6 gridpoints. With this discretization, we plot the time to solve for the approximated policy function as well as the percentage error introduced through interpolation. Increasing the number of gridpoints, we trace out the time-error frontier. The policy function method, described in Column (3), sidesteps this tradeoff entirely by solving for the exact policy function quickly. It worth pointing out that while the times recorded in the Table for the discretized single-agent problem correspond to a relatively coarse grid (21 points) an accurate approximation of the true policy function with this method implies a time increase of an order of magnitude or more.

FIGURE 7: SOLVING FOR THE POLICY FUNCTION

Countries	(1) Naive	(2) Single	(3) Policy
5	0.02	0.01	0.02
10	0.62	0.04	0.00
15	26.28	0.62	0.02
20	846.8	37.37	0.05



Notes: The table shows the time to solve (s) for the policy function mapping firm type to optimal decision set in the model of Tintelnot (2017) with varying numbers of countries. Columns 1 and 2 show trials where the productivity distribution of firms is discretized with 81 productivity gridpoints; Column 1 uses a naive brute force approach while 2 uses our single-agent method. Column 3 solves directly for the policy function using the method outlined in this paper. The figure compares the discretized single-agent method with a varying number of gridpoints against the policy function method.

5.2. Input Sourcing and Value Chain Problems

We finally describe input sourcing problems where the set of locations where the firm needs to choose to buy inputs from are chosen and corresponding value chain problems where firms produce a good that may be completed in several stages. After introducing our framework, we discuss its relation to existing frameworks and the application of our solution method.

Framework We continue to assume that the economy contains a discrete set of production locations $j = 1, \ldots, J$. We consider the sourcing problem of a firm producing a differentiated final good that must be completed in a series of sequential stages, labelled $k = 1, \ldots, K$. Each stage, each firm produces a continuum of goods which will be used as intermediates in the following stage. Intermediates in the first stage are completed using only the local factor, which we call labor and has price w_j in location j. In every subsequent stage $k > 1$, the firm requires a unit continuum of intermediates from the previous stage $k - 1$ to produce each stage-k intermediate. Each of these previous-stage intermediates are combined Cobb-Douglas with labor, with the labor share being stage-specific and denoted by $1 - \alpha^k$. The bundles are then aggregated Cobb-Douglas to form an intermediate input completed up to stage k.

In particular, consider the production of a stage-k intermediate ω^k in location j'. Indexing the necessary inputs from the previous stage $k-1$ with ω^{k-1}, the total production of a stage k intermediate is

$$q(\omega^k) = \exp\left\{ \int_{\omega^{k-1}} \ln\left[\left(\frac{q(\omega^{k-1})}{d_{j(\omega^{k-1})j'}\alpha^k} \right)^{\alpha^k} \left(\frac{\ell(\omega^{k-1})}{1-\alpha^k} \right)^{1-\alpha^k} \right] \mathrm{d}\omega^{k-1} \right\}.$$

The firm ships a quantity $q(\omega^{k-1})$ of each of the continuum of intermediate inputs from its production location $j(\omega^{k-1})$ to the production location of stage k, j' and incurs an iceberg transportation cost $d_{j(\omega^{k-1})j'} \geq 1$ in the process. The firm chooses an amount of labor $\ell(\omega^{k-1})$ to combine with this intermediate. Having bundled each intermediate ω^{k-1} with labor, the firm aggregates over these to produce a completed stage-k intermediate. Just as in the previous stage $k-1$, the firm produces a continuum of these stage-k intermediates.

After completing the intermediates from the last stage K, the firm ships them to the destination market, where they are aggregated into the firm's differentiated final good. We assume that, for each intermediate ω^K, the shipping cost is subject to a shock $\nu(\omega^K)$, which we discuss in more detail below. Thus, the total output produced for destination market m is

$$q_m = \exp\left\{ \int_{\omega^K} \ln\left[\nu(\omega^K) \frac{q_m(\omega^K)}{d_{j(\omega^K)m}} \right] \mathrm{d}\omega^K \right\}.$$

where $q_m(\omega^k)$ denotes the production of intermediate ω^k ultimately intended for market m. In what follows, we call the aggregation in the destination market stage $K+1$ in a slight abuse of notation. Under this convention, the stage $K+1$ share on labor $1-\alpha^{K+1} = 0$ since no labor from the destination market is required.

Having formalized the production process, we pause here to discuss its overall structure. The firm in our framework produces a unit continuum of intermediates every stage, each of these requiring a unit continuum of intermediates from the previous stage. In the language of Baldwin and Venables (2013), the production structure features "spider-like" elements, where each intermediate requires several inputs for production. Moreover, we do not impose that the required unit continuum of previous-stage inputs is the same for two intermediates produced in the same stage. Each intermediate may consequently have an input bundle either specific to its production or shared with other intermediates of the same stage, lending considerable flexibility to the overall production structure. A simplified version of our formulation is one where each intermediate requires a single input from the previous stage. We discuss this isomorphism in the Appendix. In this scenario, the firm carries out a "snake"-like production process for every input, aggregating them into the final good in the destination market.

The firm must choose the locations $\{j^k(\omega^k)\}_k$ in which to produce each intermediate ω^k, which we call its *sourcing strategy*. For analytical tractability, we assume that the shipping shocks $\nu(\omega^K)$ are distributed Fréchet with shape θ and are drawn for each potential sourcing strategy. The firm thus selects, for each stage-K intermediate, a sourcing strategy by minimizing unit costs of production conditional on these shocks. Since there is a continuum of stage-K intermediates, the Fréchet distribution is exactly realized. We show in the Appendix that the unit costs of production $a_m(\{\mathcal{J}^k\}_k)$ are therefore

$$a_m(\{\mathcal{J}^k\}_k) = \left[\sum_{j_m^1 \in \mathcal{J}^1} \cdots \sum_{j_m^K \in \mathcal{J}^K} \left(\prod_{k=1}^{K} w_{j_m^k}^{-\theta\beta^k} d_{j_m^k j_m^{k+1}}^{-\theta \frac{\alpha^{k+1}}{1-\alpha^{k+1}} \beta^{k+1}}\right)\right]^{-\frac{1}{\theta}} \Gamma \equiv \Theta_m(\{\mathcal{J}^k\}_k)^{-\frac{1}{\theta}} \Gamma$$

$$\beta^k = (1-\alpha^k) \prod_{i=k+1}^{K} \alpha^i \quad , \quad j_m^{K+1}(\omega) \equiv m$$

where Γ is a constant of integration and each $\mathcal{J}^k$ represents the set of available locations for the firm to carry out stage k activities. Each β^k represents the overall weight on stage-k labor, which accumulates the labor weights along the production value chain to reflect that an intermediate input from stage k is subsequently transformed stage after stage.[29]

Observe that $\Theta_m(\{\mathcal{J}^k\}_k)$ captures the cost effectiveness of a firm's set of locations $\{\mathcal{J}^k\}_k$ in serving a particular destination market m. As such, we call it the "network potential" of the firm's network. While this term is related to similar market access terms common in the gravity literature, it crucially captures network interactions between locations for the firm that are generally abset in models featuring only a single stage of production.

To more clearly highlight the role of the firm's network in the network potential, we write it in matrix form. We define the matrices $\mathbf{T}^k$ as $|\mathcal{J}^k| \times |\mathcal{J}^{k+1}|$ with (j^k, j^{k+1})th entry: $w_{j^k}^{-\theta\beta^k} d_{j^k j^{k+1}}^{-\theta \frac{1-\alpha^k}{\alpha^k} \beta^k}$. Then, the network potential can be written as the matrix product

$$\Theta_m(\{\mathcal{J}^k\}_k) = \mathbf{1}'\mathbf{T}^1\mathbf{T}^2 \ldots \mathbf{T}^K,$$

where $\mathbf{1}$ is the column vectors of ones. For each matrix $\mathbf{T}^k$, its rows represent the location of that stage while its columns represent the location of the next stage. As each good must be completed

[29] We normalize $\alpha^{K+1}\beta^{K+1}/(1-\alpha^{K+1}) = 1$. Trivially, the location of stage $K+1$ is the destination market m.

sequentially, the full matrix product captures all possible paths a given first-stage intermediate could take. With this formulation, it becomes clear that every matrix not only incorporates the cost effectiveness of a location w_j but also its proximity to each location in the firm's network as represented by the bilateral costs of transportation $d_{j^k j^{k+1}}$. Transportation costs grow costlier in later stages of production compared to earlier stages, as the fraction lost constitutes value added from many more previous stages. Similarly, high factor costs have a larger impact on the costs of production in later stages, since the local factor accounts for a larger share of the final good than in earlier stages.

Finally, the firm selects its network strategy, the set of locations $\{\mathcal{J}^k\}_k$ available at each stage. We assume that, in order to ready location j for stage k production, the firm must pay a fixed cost f_j^k. Then, the firm makes $K \times |J|$ binary choices: whether or not to prepare each location $j \in J$ for stage k production. Optimally choosing a network strategy constitutes a CDCP with objective function

$$\pi(\{\mathcal{J}^k\}_k) = \sum_m \frac{1}{\sigma} X_m \left(\frac{\sigma}{\sigma - 1} \frac{a_m(\{\mathcal{J}^k\}_k)}{P_m} \right)^{1-\sigma} - \sum_j \sum_k \mathbb{1}[j \in \mathcal{J}^k] f_j^k$$

at the firm level. This setup permits reasonable flexibility on whether location setup costs occur on the location level or the location-stage level. For example, suppose instead that a location can be used for any stage k once it is included by the firm. Then, the firm makes $|J|$ binary decisions: for each location j, whether or not to include it in the network strategy.

Single Crossing Differences Conditions In this section we discuss the single crossing conditions in the context of the general input sourcing and value chain framework introduced above. We start with a description of the fundamental economic interactions between the locations, then show how the single crossing differences conditions are related to parameter restrictions in the framework.

To begin, notice that the network potential term $\Theta_m(\{\mathcal{J}^k\}_k)$ exhibits SCD-C from below as the marginal value of a location increases as a firm grows its network: if an additional location is added to a small network, it does not create as many new sourcing strategies as if it were added to a large network. However, the firm does not choose a network strategy to maximize network potential, but firm profits. Consequently, we now turn our attention to the firm's profit function, which incorporates the network potential with an exponent $(\sigma - 1)/\theta$. This ratio, masks two additional effects that are summarized by each parameter.

On the one hand, the θ in the exponent's denominator captures the cannibalization effect among sourcing strategies, comparable to the one discussed in the plant location framework above: intuitively, additional options for sourcing strategies are more valuable when there are initially fewer sourcing strategies. As the firm expands its network, creating more sourcing strategies, it becomes less likely to choose any given one. When θ is high, shocks are more homogeneous and sourcing strategies are more likely to yield similar cost, leading to stronger cannibalization forces among them. On the other hand, the $\sigma - 1$ captures the curvature in consumer demand. When σ is high, consumers easily substitute the final goods of different firms. In such an environment, consumers are very price sensitive and, crucially, declines in the firm's unit cost creates larger profit gains for firms with already low prices. This effect implies that, as the firm's network grows and its prices fall in response, an additional sourcing strategy becomes more valuable.

Taken together, the network effect and the curvature of consumer deman dimply positive complementarities while the cannibalization effect implies negative. Thus, a sufficient condition for SCD-C from below is for the consumer demand effect to overcome the cannibalization effect, corresponding to the parameter restriction $\sigma - 1 \geq \theta$. Intuitively, this parameter restriction asserts that the negative interactions from cannibalization are overpowered by the positive interactions from curvature in consumer demand.

However, if there is only one stage of production so that $K = 1$, there are no network effects. In this case, when only the effects of cannibalization and curvature in consumer demand remain, $\sigma - 1 < \theta$ is a sufficient parameter restriction to ensure SCD-C from above.

Finally, if $\sigma - 1 < \theta$ while there are multiple stages of production so that $K > 1$, the direction of complementarities is ambiguous and may change over the firm's decision space. In particular, since the network effect incorporates both information on factor prices and geography, there is no simple parameter restriction that delivers a sufficient condition for SCD-C from above.

We conclude with a discussion of SCD-T in this framework by introducing dimensions of firm heterogeneity similarly to those in the plant location setup. In particular, we allow firms to be heterogeneous in a scalar variable cost shifter z and the fixed costs of setup f_j^k. The profit function in this scenario is now parameterized by these firm characteristics, taking the form

$$\pi(\{\mathcal{J}^k\}_k; z, \mathbf{f}) = \sum_m \frac{1}{\sigma} X_m \left(\frac{\sigma}{\sigma - 1} \frac{a_m(\{\mathcal{J}^k\}_k)}{z P_m} \right)^{1-\sigma} - \sum_j \sum_k \mathbb{1}[j \in \mathcal{J}^k] f_j^k$$

and crucially exhibiting SCD-T. Profit functions featuring a firm-specific variable cost shifter may arise under a number of assumptions. For example, z may describe the firm's Hicks-neutral productivity in every stage. In other words, fixing a production strategy, a firm of type $2z$ will produce twice as much as a firm of type z in each stage. An alternative assumption would be that shipping shocks are drawn from a Fréchet with scale z, so that high-type firms are more efficient at shipping than low-type firms. Both these environments will similarly imply that unit costs of production are ultimately shifted by the firm's type z, delivering a profit function obeying SCD-T.

Input Sourcing and Value Chain Applications As we discussed above we provide a characterization of a joint spider (input-sourcing for each stage) and snake (value chain across stages) problem.[30] We now discuss two limit cases of our model that correspond to an input sourcing and a value chain model in the literature, respectively.

Assuming that there is only the final stage, $K = 1$, and a continuum of stage-K goods are shipped to the destination market then aggregated into the final firm-specific good yields a setup similar to the influential work of Antras et al. (2017). Each firm chooses a set of countries $\mathcal{J}$ from which to source their inputs, which corresponds to $\mathcal{J}^K = \mathcal{J}^1$ in our setup. As discussed in the paper, the relative size of $\sigma - 1$ compared to θ delivers a simple parameter restriction for SCD-C from above or below. Intuitively, since there is only one stage of production, the network effects described above are absent in the firm's problem. As a result, there remain only the cannibalization and consumer demand effects, the strength of which are governed by $\sigma - 1$ and θ respectively. Thus, the firm's CDCP exhibits SCD-C from below if $\sigma - 1 > \theta$, from above if $\sigma - 1 < \theta$, and no interdependencies if $\sigma - 1 = \theta$. Firms in this setup differ on the fixed costs of setup and a core efficiency. The core productivity of the firm serves as a Hicks-neutral productivity, shifting costs in the aggregation stage. These assumptions deliver a profit function of the form discussed above, consequently exhibiting SCD-T. Since the quantitative exercises focus on the case where cannibalization forces are dominated by the effects of consumer demand, the numerical problem exhibits both SCD-C from below and SCD-T. As previously mentioned and also discussed in Antras et al. (2017), the combination of these two conditions implies that the optimal sets $\mathcal{J}(z)$ are nested. In other words, holding the fixed costs of production constant, the optimal decision sets of more productive firms necessarily contain the optimal decision sets of less productive firms.

The other polar case of our framework is the one where each stage requires a single intermediate from the previous stage, which is combined with the local factor bundle that incorporates a basket of final goods. This is related to the work of Antràs and de Gortari (2017), Johnson and Moxnes (2019) who study the geography of global value chains, emphasizing that some countries may have a comparative advantage in downstream stages while others in upstream stages. The study presents a perfectly competitive framework for Global Value Chains, where a final good is completed stage by stage. In the aggregate, production shares in our framework mirror the implications in these

[30] Assuming an increasing cost of adding additional stages of production our framework permits the possibility of value chain problems with an endogenous choice of the number of stages, as in Grant and Startz (2019). The intuition in this case is that adding an extra stage will yield increasingly positive benefits to the firm due to the aforementioned network effects. We are currently working on the characterization of the precise SCD conditions for that case.

frameworks under the Frechet assumption, but are realised over the continuum of firms rather than at the firm level. By comparison, our framework allows each stage to require a *continuum* of inputs from the previous stage, so that production shares are realized at the firm level.

Notice that we also allow firms to choose ex-ante which locations will ultimately be available for production at each level, paying a fixed cost to establish these locations. Incorporating this network strategy introduces a CDCP at the firm level that is absent from those prior works that assume perfect competition, which instead assume that all locations are available for all stages at no additional cost. [31]

6. Concluding Remarks

TBD

[31] Finally, Antràs and de Gortari (2017) explores several shock structures in their model, including both stage-level and path-level shocks, and ultimately show that these structures are isomorphic under certain assumptions. Accordingly, we impose a Fréchet shock in the last stage of production. This assumption is isomorphic to allowing stage-level shocks, as long as their product is distributed Fréchet.

References

ANTRÀS, P. AND D. CHOR (2021): "Global Value Chains," .

ANTRÀS, P. AND A. DE GORTARI (2017): "On the Geography of Global Value Chains," Harvard mimeo.

ANTRAS, P., T. C. FORT, AND F. TINTELNOT (2017): "The margins of global sourcing: Theory and evidence from us firms," *American Economic Review*, 107, 2514–64.

ARKOLAKIS, C., N. RAMONDO, A. RODRÍGUEZ-CLARE, AND S. YEAPLE (2018): "Innovation and production in the global economy," *American Economic Review*, 108, 2128–73.

BALDWIN, R. AND A. J. VENABLES (2013): "Spiders and snakes: Offshoring and agglomeration in the global economy," *Journal of International Economics*, 90, 245–254.

BALINSKI, M. L. (1965): "Integer programming: methods, uses, computations," *Management science*, 12, 253–313.

BRYAN, G. AND M. MORTEN (2019): "The aggregate productivity effects of internal migration: Evidence from indonesia," *Journal of Political Economy*, 127, 2229–2268.

COSTINOT, A. (2009): "An Elementary Theory of Comparative Advantage," *Econometrica*, 77, 1165–1192.

EATON, J. AND S. KORTUM (2002): "Technology, geography, and trade," *Econometrica*, 70, 1741–1779.

GRANT, M. AND M. STARTZ (2019): "Cutting Out the Middleman: The Structure of Chains of Intermediation," .

GUADAGNI, P. M. AND J. D. LITTLE (1983): "A logit model of brand choice calibrated on scanner data," *Marketing science*, 2, 203–238.

HENDEL, I. (1999): "Estimating multiple-discrete choice models: An application to computerization returns," *The Review of Economic Studies*, 66, 423–446.

HOLMES, T. J. (2011): "The diffusion of Wal-Mart and economies of density," *Econometrica*, 79, 253–302.

JAKOB, K. AND P. M. PRUZAN (1983): "The simple plant location problem: survey and synthesis," *European journal of operational research*, 12, 36–81.

JIA, P. (2008): "What happens when Wal-Mart comes to town: An empirical analysis of the discount retailing industry," *Econometrica*, 76, 1263–1316.

JIANG, E. AND V. TYAZHELNIKOV (2020): "Shadow O shoring and Complementarity in Inputs," .

JOHNSON, R. C. AND A. MOXNES (2019): "GVCs and trade elasticities with multistage production," Tech. rep., National Bureau of Economic Research.

KARTIK, N., S. LEE, AND D. RAPPOPORT (2019): "Single-crossing differences on distributions," Tech. rep.

LIND, N. AND N. RAMONDO (2018): "Trade with correlation," Tech. rep., National Bureau of Economic Research.

MCFADDEN, D. (1973): "Conditional logit analysis of qualitative choice behavior," .

MILGROM, P. R. (2004): *Putting Auction Theory to Work*, Cambridge University Press.

MORALES, E., G. SHEU, AND A. ZAHLER (2019): "Extended gravity," *The Review of Economic Studies*, 86, 2668–2712.

OWEN, S. H. AND M. S. DASKIN (1998): "Strategic facility location: A review," *European journal of operational research*, 111, 423–447.

PAKES, A., J. PORTER, K. HO, AND J. ISHII (2015): "Moment inequalities and their application," *Econometrica*, 83, 315–334.

RAMONDO, N. (2014): "A quantitative approach to multinational production," *Journal of International Economics*, 93, 108–122.

RAMONDO, N. AND A. RODRÍGUEZ-CLARE (2013): "Trade, multinational production, and the gains from openness," *Journal of Political Economy*, 121, 273–322.

TINTELNOT, F. (2016): "Global production with export platforms," *The Quarterly Journal of Economics*, qjw037.

——— (2017): "Global production with export platforms," *The Quarterly Journal of Economics*, 132, 157–209.

TOPKIS, D. M. (1978): "Minimizing a submodular function on a lattice," *Operations research*, 26, 305–321.

TRAIN, K. (1986): *Qualitative choice analysis: Theory, econometrics, and an application to automobile demand*, vol. 10, MIT press.

YANG, C. (2020): "Location choices of multi-plant oligopolists: Theory and evidence from the cement industry," .

ZHENG, F. (2016): "Spatial competition and preemptive entry in the discount retail industry," *Columbia Business School Research Paper*.

Appendix

A. Single crossing difference conditions

In this section, we discuss the single crossing difference conditions from the main text. We begin by clarifying the relationship between sub- and super-modularity with monotonicity in set. We show that the first pair of conditions are sufficient for the second; we then show that the second are sufficient for the first when the choice space is finite. A short discussion follows outlining a counterexample for the case of an infinite choice space. Finally, we verify the sufficient condition for SCD-T provided in the main text.

In this first proposition, we show that submodularity and supermodularity are sufficient to ensure monotonicity in set.

Proposition (Sufficiency of sub- and super-nmodularity)**.** *Fix* $\mathbf{z}$ *and* $\mathbf{y}$ *and consider the return function* π*.*

- *If* π *is submodular, then* π *exhibits monotonicity in set of the substitutes form.*
- *If* π *is supermodular, then* π *exhibits monotonicity in set of the complements form.*

Proof. Since $\mathbf{z}$ and $\mathbf{y}$ are fixed during this proof, they are omitted for notational brevity.

Begin with π submodular. Then, for any sets A, B, it is the case that

$$\pi(A) + \pi(B) \geq \pi(A \cup B) + \pi(A \cap B)\,.$$

We show that monotonicity in set of the substitutes form must hold. Let $\mathcal{J}_1 \subseteq \mathcal{J}_2$. Select an arbitrary j. The goal is to show that

$$\begin{aligned}
\pi(\mathcal{J}_1 \cup \{j\}) - \pi(\mathcal{J}_1) &\geq \pi(\mathcal{J}_2 \cup \{j\}) - \pi(\mathcal{J}_2) && \text{if } j \notin \mathcal{J}_2\text{, so } j \notin \mathcal{J}_1 \\
\pi(\mathcal{J}_1 \cup \{j\}) - \pi(\mathcal{J}_1) &\geq \pi(\mathcal{J}_2) - \pi(\mathcal{J}_2 \setminus \{j\}) && \text{if } j \in \mathcal{J}_2\text{, but } j \notin \mathcal{J}_1 \\
\pi(\mathcal{J}_1) - \pi(\mathcal{J}_1 \setminus \{j\}) &\geq \pi(\mathcal{J}_2) - \pi(\mathcal{J}_2 \setminus \{j\}) && \text{if } j \in \mathcal{J}_1\text{, so } j \in \mathcal{J}_2
\end{aligned}$$

Define the sets A and B as below for each corresponding scenario.

$$\begin{aligned}
&A \equiv \mathcal{J}_1 \cup \{j\} && B \equiv \mathcal{J}_2 && \text{if } j \notin \mathcal{J}_2\text{, so } j \notin \mathcal{J}_1 \\
&A \equiv \mathcal{J}_1 \cup \{j\} && B \equiv \mathcal{J}_2 \setminus \{j\} && \text{if } j \in \mathcal{J}_2\text{, but } j \notin \mathcal{J}_1 \\
&A \equiv \mathcal{J}_1 && B \equiv \mathcal{J}_2 \setminus \{j\} && \text{if } j \in \mathcal{J}_1\text{, so } j \in \mathcal{J}_2
\end{aligned}$$

Then, it is easy to see that applying the submodularity condition implies monotonicity in set of the substitutes form.

Now, suppose π is supermodular. Then, for any sets A, B, it is the case that

$$\pi(A) + \pi(B) \leq \pi(A \cup B) + \pi(A \cap B) .$$

We show that monotonicity in set of the complements form must hold. Let $\mathcal{J}_1 \subseteq \mathcal{J}_2$. Select an arbitrary j. The goal is to show that

$$\begin{aligned} \pi(\mathcal{J}_1 \cup \{j\}) - \pi(\mathcal{J}_1) &\leq \pi(\mathcal{J}_2 \cup \{j\}) - \pi(\mathcal{J}_2) && \text{if } j \notin \mathcal{J}_2, \text{ so } j \notin \mathcal{J}_1 \\ \pi(\mathcal{J}_1 \cup \{j\}) - \pi(\mathcal{J}_1) &\leq \pi(\mathcal{J}_2) - \pi(\mathcal{J}_2 \setminus \{j\}) && \text{if } j \in \mathcal{J}_2, \text{ but } j \notin \mathcal{J}_1 \\ \pi(\mathcal{J}_1) - \pi(\mathcal{J}_1 \setminus \{j\}) &\leq \pi(\mathcal{J}_2) - \pi(\mathcal{J}_2 \setminus \{j\}) && \text{if } j \in \mathcal{J}_1, \text{ so } j \in \mathcal{J}_2 \end{aligned}$$

Define the sets A and B identically as above for each corresponding scenario. Then, it is easy to see that applying the supermodularity implies monotonicity in set of the complements form. □

In this second proposition, we show that, conditional on a finite choice space J, monotonicity in set implies sub- or super-modularity.

Proposition (Sufficiency of monotonicity in set with finite choice space)**.** *Fix* $\mathbf{z}$ *and* $\mathbf{y}$ *and consider the return function* π*. Let* A *and* B *be arbitrary sets so that* $A \setminus (A \cap B)$ *is finite.*

- *If* π *exhibits monotonicity in set of the substitutes form, then*

$$\pi(A; \mathbf{z}, \mathbf{y}) + \pi(B; \mathbf{z}, \mathbf{y}) \geq \pi(A \cup B; \mathbf{z}, \mathbf{y}) + \pi(A \cap B; \mathbf{z}, \mathbf{y}) .$$

- *If* π *exhibits monotonicity in set of the complements form, then*

$$\pi(A; \mathbf{z}, \mathbf{y}) + \pi(B; \mathbf{z}, \mathbf{y}) \leq \pi(A \cup B; \mathbf{z}, \mathbf{y}) + \pi(A \cap B; \mathbf{z}, \mathbf{y}) .$$

Proof. Since $\mathbf{z}$ and $\mathbf{y}$ are fixed during this proof, they are omitted for notational brevity. Let $\tilde{A}$ and $\tilde{B}$ be arbitrary sets where $\tilde{A} \setminus (\tilde{A} \cap \tilde{B})$ is finite.

Begin with monotonicity in set of the substitutes form first. Define

$$I \equiv \tilde{A} \cap \tilde{B} \qquad A \equiv \tilde{A} \setminus I \qquad B \equiv \tilde{B} \setminus I .$$

Then, it is equivalent to show that

$$\pi(I \cup A) + \pi(I \cup B) \geq \pi(I) + \pi(I \cup A \cup B)\,. \tag{A.1}$$

The proof proceeds inductively on the cardinality of A. When A is empty, then (A.1) holds with equality.

Now suppose (A.1) holds for $|A| = n$. Consider the case where $|A| = n + 1$. Let a be an arbitrary element from A and define $\underline{A} \equiv A \setminus \{a\}$ as A with a excluded. From the inductive assumption,

$$\pi(I) + \pi(I \cup \underline{A} \cup B) \leq \pi(I \cup \underline{A}) + \pi(I \cup B)$$

while from monotonicity in set of the substitutes form,

$$\begin{gathered} D_a\pi(I \cup \underline{A} \cup B) \leq D_a\pi(I \cup \underline{A}) \\ \pi(I \cup A \cup B) - \pi(I \cup \underline{A} \cup B) \leq \pi(I \cup A) - \pi(I \cup \underline{A})\,. \end{gathered}$$

Combining the two expressions together yields

$$\pi(I) + \pi(I \cup A \cup B) \leq \pi(I \cup A) + \pi(I \cup A)\,,$$

which confirms (A.1) for sets A of cardinality $n + 1$. The inductive proof establishes that (A.1) holds for all A of finite size.

Next, consider monotonicity in set of the complements form. The argument follows a similar structure. Now, it is equivalent to show that

$$\pi(I \cup A) + \pi(I \cup B) \leq \pi(I) + \pi(I \cup A \cup B)\,. \tag{A.2}$$

Proceed inductively once again on the cardinality of A. When A is empty, (A.2) holds with equality. Now suppose (A.2) holds for A with cardinality n. Consider A with cardinality $n + 1$. Similarly, select an arbitrary element $a \in A$ and define $\underline{A} \equiv A \setminus \{a\}$. The inductive assumption implies that

$$\pi(I) + \pi(I \cup \underline{A} \cup B) \geq \pi(I \cup \underline{A}) + \pi(I \cup B)$$

while from monotonicity in set of the complements form,

$$\begin{gathered} D_a\pi(I \cup \underline{A} \cup B) \geq D_a\pi(I \cup \underline{A}) \\ \pi(I \cup A \cup B) - \pi(I \cup \underline{A} \cup B) \geq \pi(I \cup A) - \pi(I \cup \underline{A})\,. \end{gathered}$$

Combining the two expressions together yields

$$\pi(I) + \pi(I \cup A \cup B) \geq \pi(I \cup A) + \pi(I \cup A)\,,$$

which confirms (A.2) for sets A of cardinality $n+1$. The inductive proof establishes that (A.2) holds for all A of finite size. $\square$

When $A \setminus (A \cap B)$ is not finite, then monotonicity in set does not necessarily ensure supermodularity or submodularity. As a simple counterexample, suppose the return π of a decision set S is defined

$$\pi(S) = \left[\int_S 1 \,\mathrm{d}s\right]^\alpha$$

and note that the marginal value of any item j follows as

$$D_j\pi(S) = \left[\int_{S\cup\{j\}} 1 \,\mathrm{d}s\right]^\alpha - \left[\int_{S\setminus\{j\}} 1 \,\mathrm{d}s\right]^\alpha = 0\,.$$

The intuition is simple: since we integrate over the a decision set S for its return, any singular element j is measure zero and has no effect on the decision set's overall return. The return function therefore satsifies monotonicity in set (of both forms).

Now consider $A = [0,2]$ and $B = [1,3]$. It is easy to see that

$$\begin{aligned} \pi(A) &= 2^\alpha & \qquad \pi(A \cup B) &= 3^\alpha \\ \pi(B) &= 2^\alpha & \qquad \pi(A \cap B) &= 1^\alpha \end{aligned}$$

so, in this case,

$$\begin{aligned} \alpha > 1 \qquad &\Rightarrow \qquad \pi(A) + \pi(B) > \pi(A \cup B) + \pi(A \cap B) \\ \alpha \in (0,1) \qquad &\Rightarrow \qquad \pi(A) + \pi(B) < \pi(A \cup B) + \pi(A \cap B)\,. \end{aligned}$$

Then, when $\alpha > 1$, the return function obeys monotonicity in set of the substitutes form but violates submodularity. Likewise, when $\alpha \in (0,1)$, the return function obeys monotonicity in set of the complements form but violates supermodularity.

In this next proposition, we establish the sufficient condition for SCD-T provided in the main body of the paper.

Proposition (Sufficient condition for SCD-T). *Fix an item j and $\mathcal{J}$. Let the entries of $\mathbf{z}$ be indexed by i,* *so that z_i is the ith coordinate of* $\mathbf{z}$*. Suppose*

$$\frac{\partial D_j \pi(\mathcal{J};\mathbf{z})}{\partial z_i}$$

(weakly) maintains the same sign over the entire typespace for each coordinate i. Then, the problem exhibits SCD-T.

Proof. We first show that $Z_j^+(\mathcal{J})$ is a path-connected, and thus connected, set. Let $\mathbf{z}$ and $\mathbf{z}'$ both be in the set. The proof proceeds by constructing a path from $\mathbf{z}$ to $\mathbf{z}'$. First, we construct the point $\tilde{\mathbf{z}}$ where

$$\tilde{z}_i = \begin{cases} \max\{z_i, z_i'\} & \text{if } \frac{\partial D_j \pi(\mathcal{J};\mathbf{z})}{\partial z_i} \geq 0 \\ \min\{z_i, z_i'\} & \text{if } \frac{\partial D_j \pi(\mathcal{J};\mathbf{z})}{\partial z_i} \leq 0 \end{cases}.$$

Gather the indices $I \equiv \{i \mid z_i \neq \tilde{z}_i\}$. Index them from $m = 1$ to $m = |I|$ and construct the sequence of points $\{\mathbf{z}_0, \mathbf{z}_1, \ldots, \mathbf{z}_m, \ldots, \mathbf{z}_{|I|}\}$ where

$$\mathbf{z}_0 = \mathbf{z} \qquad\qquad \mathbf{z}_m = \mathbf{z}_{m-1} + \mathbf{j}_{i_m}(\tilde{z}_{i_m} - z_{i_m})$$

and $\mathbf{j}_i$ is the ith standard basis vector (that is, the vector with 1 in the ith coordinate and 0 everywhere else). At each step of the sequence, the i_mth coordinate is changed to $\tilde{z}_{i_m}$ and all other coordinates are unchanged. Then, we construct the piece-wise linear path from $\mathbf{z}$ to $\tilde{\mathbf{z}}$ sequentially passing through these points.

This path is contained in $Z_j^+(\mathcal{J})$ by construction. In particular, $D_j\pi(\mathcal{J};\cdot)$ starts positive on this path by assumption on $\mathbf{z}$. In each mth segment of the path, only the i_mth component changes while all others stay constant. If the partial derivative of $D_j\pi(\mathcal{J};\cdot)$ along this dimension is (weakly) positive, the coordinate is increased; otherwise, it is decreased. Thus, $D_j\pi(\mathcal{J};\cdot)$ weakly increases along the path, and so cannot ever fall below zero.

We similarly construct a piece-wise linear path from $\mathbf{z}'$ to $\tilde{\mathbf{z}}$ that lies in $Z_j^+(\mathcal{J})$. Joining these paths together at $\tilde{\mathbf{z}}$, we have constructed a path from $\mathbf{z}$ to $\mathbf{z}'$ that remains in $Z_j^+(\mathcal{J})$. Since $\mathbf{z}$ and $\mathbf{z}'$ were any arbitrary members of $Z_j^+(\mathcal{J})$, we have shown that it is path-connected. Showing $Z_j^-(\mathcal{J})$ is path-connected follows a similar argument. Now suppose $\mathbf{z}$ and $\mathbf{z}'$ are contained in $Z_j^-(\mathcal{J})$. We construct $\tilde{\mathbf{z}}$ in this case as

$$\tilde{z}_i = \begin{cases} \max\{z_i, z_i'\} & \text{if } \frac{\partial D_j \pi(\mathcal{J};\mathbf{z})}{\partial z_i} \leq 0 \\ \min\{z_i, z_i'\} & \text{if } \frac{\partial D_j \pi(\mathcal{J};\mathbf{z})}{\partial z_i} \geq 0 \end{cases}.$$

We then construct the paths from $\mathbf{z}$ to $\tilde{\mathbf{z}}$ and from $\mathbf{z}'$ to $\tilde{\mathbf{z}}$ in the same way. Both lie in $Z_j^-(\mathcal{J})$ by the same logic. □

B. Solution results

In this section, we discuss results related to our solution methods and the policy function. We begin by proving Theorems 1 and 2 before closing with characterisations of the policy function under SCD-C.

We start with Theorem 1.

Proof. Suppose the return function satisfies SCD-C from above. We prove inductively that successively applying the squeezing step weakly narrows down the choice space without eliminating the optimal decision set. Let the bounding sets after the nth application be $[\underline{\mathcal{J}}^{(k)}, \overline{\mathcal{J}}^{(k)}]$.

Starting with $[\emptyset, J]$, it is trivially the case that $\emptyset \subseteq \underline{\mathcal{J}}^{(1)}$ and $\overline{\mathcal{J}}^{(1)} \subseteq J$. What remains to show is that this first pair of bounding sets sandwiches $J^\star$. Let $j \in \underline{\mathcal{J}}^{(1)}$. Then, $D_j\pi(J) > 0$ so $D_j\pi(\mathcal{J}^\star) > 0$ by SCD-C from above. So $j \in \mathcal{J}^\star$. Since j was an arbitrary element of $\underline{\mathcal{J}}^{(1)}$, we conclude $\underline{\mathcal{J}}^{(1)} \subseteq \mathcal{J}^\star$. Next, consider $j \in \mathcal{J}^\star$. Then, $D_j\pi(\mathcal{J}^\star) > 0$ by optimality, so $D_j\pi(\emptyset) > 0$ by SCD-C from above. Thus, $j \in \overline{\mathcal{J}}^{(1)}$ and since j was an arbitrary member of $\mathcal{J}^\star$, it must be the case that $\mathcal{J}^\star \subseteq \underline{\mathcal{J}}^{(1)}$.

Now suppose $\underline{\mathcal{J}}^{(n-1)} \subseteq \underline{\mathcal{J}}^{(n)} \subseteq \mathcal{J}^\star \subseteq \overline{\mathcal{J}}^{(n)} \subseteq \overline{\mathcal{J}}^{(n-1)}$. We show that $\underline{\mathcal{J}}^{(n)} \subseteq \underline{\mathcal{J}}^{(n+1)} \subseteq \mathcal{J}^\star \subseteq \overline{\mathcal{J}}^{(n+1)} \subseteq \overline{\mathcal{J}}^{(n)}$. Select $j \in \underline{\mathcal{J}}^{(n)}$. It must be the case that $D_j\pi(\overline{\mathcal{J}}^{(n-1)}) > 0$. Since $\overline{\mathcal{J}}^{(n)} \subseteq \overline{\mathcal{J}}^{(n-1)}$, SCD-C from above implies $D_j\pi(\overline{\mathcal{J}}^{(n)}) > 0$, so $j \in \underline{\mathcal{J}}^{(n+1)}$. Since j was an arbitrary element of $\underline{\mathcal{J}}^{(n)}$, we conclude $\underline{\mathcal{J}}^{(n)} \subseteq \underline{\mathcal{J}}^{(n+1)}$. Similarly, now select $j \in \mathcal{J}^{(n+1)}$. We show it is in $\mathcal{J}^{(n)}$. Because $D_j(\underline{\mathcal{J}}^{(n)}) > 0$ and $\underline{\mathcal{J}}^{(n-1)} \subseteq \underline{\mathcal{J}}^{(n)}$, SCD-C from above ensures that that $D_j(\underline{\mathcal{J}}^{(n-1)}) > 0$. We conclude $\mathcal{J}^{(n+1)} \subseteq \mathcal{J}^{(n)}$.

We now show $\underline{\mathcal{J}}^{(n+1)}$ and $\overline{\mathcal{J}}^{(n+1)}$ sandwich the optimal decision set. Let $j \in \underline{\mathcal{J}}^{(n+1)}$ so that $D_j(\overline{\mathcal{J}}^{(n)}) > 0$. By the inductive assumption, $\mathcal{J}^\star \subseteq \mathcal{J}^{(n)}$, so SCD-C from above allows us to conclude that $D_j(\mathcal{J}^\star) > 0$, implying $j \in \mathcal{J}^\star$ optimally. Similarly, suppose $j \in \mathcal{J}^\star$ so that $D_j(\mathcal{J}^\star) > 0$. By the inductive assumption, $\underline{\mathcal{J}}^{(n)} \subseteq \mathcal{J}^\star$, so SCD-C from above implies $D_j(\underline{\mathcal{J}}^{(n)}) > 0$, ensuring $j \in \underline{\mathcal{J}}^{(n+1)}$.

A similar argument follows for the case where SCD-C from below holds.

Finally, the squeezing procedure must complete in under $|J|$ iterations. Each iteration, if no new items are fixed, then the procedure has converged. As a result, it must be that each iteration fixes at least one item if the procedure continues. Thus, the maximal number of iterations is achieved when exactly one item is fixed each time, with all $|J|$ items eventually being fixed. $\square$

Having proven Theorem 1, the proof for Theorem 2 follows similarly.

Proof. Consider the 4-ple $[(\underline{\mathcal{J}}, \overline{\mathcal{J}}, M), Z]$ and suppose the return function obeys SCD-C from above and SCD-T. We show that the generalized squeezing step exhaustively partitions Z into disjoint subregions, so that the new 4-ples induce functions $\underline{\mathcal{J}}(\cdot)$ and $\overline{\mathcal{J}}(\cdot)$ over Z. We then show $\underline{\mathcal{J}} \subseteq \underline{\mathcal{J}}(\mathbf{z}) \subseteq \mathcal{J}^\star(\mathbf{z}) \subseteq \overline{\mathcal{J}}(\mathbf{z}) \subseteq \overline{\mathcal{J}}$ for every $\mathbf{z} \in Z$.

First, observe that $Z_j^+(\overline{\mathcal{J}})$ and $Z_j^-(\underline{\mathcal{J}})$ are disjoint. For any type $\mathbf{z}$ receiving positive benefit from j's addition to $\overline{\mathcal{J}}$, SCD-C from above implies that it must receive positive benefit from j's addition to $\underline{\mathcal{J}}$. Then, $Z_j^0(\underline{\mathcal{J}}) \cup Z_j^+(\underline{\mathcal{J}})$ is the complement of $Z_j^-(\underline{\mathcal{J}})$, from which $Z_j^-(\underline{\mathcal{J}})$ has been removed for the third 4-ple. The three new 4-ples thus induce a valid partitioning on Z, inducing the functions $\underline{\mathcal{J}}(\cdot)$ and $\overline{\mathcal{J}}(\cdot)$.

Next, it is trivially the case that $\underline{\mathcal{J}} \subseteq \underline{\mathcal{J}}(\mathbf{z})$ for all $\mathbf{z} \in Z$ since $\underline{\mathcal{J}}(\mathbf{z})$ is either $\underline{\mathcal{J}}$ or $\underline{\mathcal{J}} \cup \{j\}$. A similar argument establishes that $\overline{\mathcal{J}}(\mathbf{z}) \subseteq \overline{\mathcal{J}}$ for all $\mathbf{z} \in Z$. We now show that $\underline{\mathcal{J}}(\mathbf{z}) \subseteq \mathcal{J}^\star(\mathbf{z}) \subseteq \overline{\mathcal{J}}(\mathbf{z})$ for all $\mathbf{z} \in Z$. Consider an element from $\underline{\mathcal{J}}(\mathbf{z})$. It is either an element from $\underline{\mathcal{J}}_0$, which is a subset of $\mathcal{J}^\star(\mathbf{z})$ by assumption, or it is j. In particular, j is in $\underline{\mathcal{J}}(\mathbf{z})$ only for $\mathbf{z} \in Z_j^+(\overline{\mathcal{J}})$. These are the types in $\mathbf{z}$ deriving positive marginal value of j's addition to $\overline{\mathcal{J}}$. For these types $\mathbf{z}$, $D_j\pi(\mathcal{J}^\star(\mathbf{z}), \mathbf{z}) > 0$ by SCD-C from above, since $\mathcal{J}^\star(\mathbf{z}) \subseteq \overline{\mathcal{J}}$ by assumption. Now, we show that $\mathcal{J}^\star(\mathbf{z}) \subseteq \overline{\mathcal{J}}(\mathbf{z})$ for all $\mathbf{z} \in Z$. Note that $\mathcal{J}^\star(\mathbf{z}) \subseteq \overline{\mathcal{J}}_0$, but assumption. Further, $\overline{\mathcal{J}}(\mathbf{z}) = \overline{\mathcal{J}}_0$ for all 4-ples except the second, which is associated with subregion $Z_j^-(\underline{\mathcal{J}})$. For types $\mathbf{z}$ in this subregion, $\overline{\mathcal{J}}(\mathbf{z}) = \overline{\mathcal{J}}_0 \setminus \{j\}$. What remains to show, then, is that $j \notin \mathcal{J}^\star(\mathbf{z})$ for types in this subregion. By definition, $D_j\pi(\underline{\mathcal{J}}, \mathbf{z}) < 0$ for types in this region, so it must be that $D_j(\mathcal{J}^\star(\mathbf{z}), \mathbf{z}) \leq 0$ by SCD-C from above. We therefore conclude that $j \notin \mathcal{J}^\star(\mathbf{z})$ for these types.

A similar argument follows for return functions exhibiting SCD-C from below instead of SCD-C from above. $\square$

The next proposition establishes that, with SCD-C from below, SCD-T, and a single dimensional typespace, the policy function features a nesting structure.

Proposition (Nested policy function on single dimensional typespace)**.** *Consider the satisfiying SCD-C from below and SCD-T where the choice set J is finite. Then, for any $z_1 < z_2$, it must be that $\mathcal{J}^\star(z_1) \subseteq$*

$\mathcal{J}^\star(z_2)$.

Proof. For a contradiction, suppose not. Define $\mathcal{J}^o \equiv \mathcal{J}^\star(z_1) \setminus \mathcal{J}^\star(z_2)$, which has cardinality $N \in (0, \infty)$ by assumption. Further, let $\mathcal{J}^i \equiv \mathcal{J}^\star(z_1) \cap \mathcal{J}^\star(z_1)$ so that $\mathcal{J}^\star(z_1) = \mathcal{J}^i \cup \mathcal{J}^o$.

Claim: for any finite N, it must be that $\mathcal{J}^\star(z_2) \cup \mathcal{J}^o$ is preferable to $\mathcal{J}^\star(z_2)$ for z_2 types. Note $\mathcal{J}^o \neq \emptyset$, so $\mathcal{J}^\star(z_2) \cup \mathcal{J}^o \neq \mathcal{J}^\star(z_2)$. This claim represents a contradiction to $\mathcal{J}^\star(z_2)$ being the decision set for z_2. The proof for the claim proceeds by induction on N. Suppose $N = 1$. Let its element be j. Then,

$$
\begin{aligned}
0 < D_j\pi(\mathcal{J}^\star(z_1), z_1) &= D_j\pi(\mathcal{J}^i, z_1) \\
\implies 0 &< D_j\pi(\mathcal{J}^i, z_2) \\
\implies 0 &< D_j\pi(\mathcal{J}^\star(z_2), z_2) \\
\implies \pi(\mathcal{J}^\star(z_2) \cup \{j\}, z_2) &> \pi(\mathcal{J}^\star(z_2), z_2)
\end{aligned}
$$

where the first line derives from z_1 optimality[32], the second from single-dimensional SCDT, and the third SCD-C from above, and the fourth from recognising that $j \notin \mathcal{J}^\star(z_2)$. This argument proves the claim for $N = 1$.

Now suppose the claim holds for $N = n$. We now show it holds for $n + 1$. Select any item $j \in \mathcal{J}^o$ and let $J_n^o \equiv \mathcal{J}^o \setminus \{j\}$. Then, by the inductive assumption,

$$
\pi(\mathcal{J}^\star(z_2), z_2) < \pi(\mathcal{J}^\star(z_2) \cup \mathcal{J}_n^o, z_2)\,.
$$

Using the optimality of z_1 types, SCDT, and SCDC-B,

$$
\begin{aligned}
0 < D_j\pi(\mathcal{J}^\star(z_1), z_1) &= D_j\pi(\mathcal{J}^i \cup \mathcal{J}_n^o; z_1) \\
\implies 0 &< D_j\pi(\mathcal{J}^i \cup \mathcal{J}_n^o; z_2) \\
\implies 0 &< D_j(\mathcal{J}^\star(z_2) \cup J_n^o; z_2) \\
\implies \pi(\mathcal{J}^\star(z_2) \cup \mathcal{J}_n^o \cup \{j\}, z_2) &> \pi(\mathcal{J}^\star(z_2) \cup \mathcal{J}_n^o, z_2) \geq \pi(\mathcal{J}^\star(z_2), z_2)
\end{aligned}
$$

where the last inequality makes use of the inductive assumption. Thus we show the claim holds for $n + 1$ if it holds for n.

We have shown that, for any finite non-empty $\mathcal{J}^o$, agents of type z_2 prefer $\mathcal{J}^\star(z_2) \cup \mathcal{J}^o$ to $\mathcal{J}^\star(z_2)$, a contradiction. □

[32] To break ties, we assume that all items for which an agent is indifferent are excluded from the the optimal set. These are easily identified and can be included into optimal sets if desired.

We extend the nesting result to the multidimensional setting in this next proposition, with a stricter condition in place of SCD-T.

Proposition (Nested policy function on multidimensional typespace). *Suppose that the return function satisfies SCD-C from below and the choice set J is finite. In addition, suppose that, for each component i of the type vector,*

$$\frac{\partial D_j \pi(\mathcal{J}; \mathbf{z})}{\partial z_i}$$

maintains the same sign for all j, $\mathcal{J}$, and $\mathbf{z}$. *Consider two types $\mathbf{z}_1$ and $\mathbf{z}_2$ where the ith component of $\mathbf{z}_2 - \mathbf{z}_1$ has the same sign as the ith partial derivative above. Then, $\mathcal{J}^\star(\mathbf{z}_1) \subseteq \mathcal{J}^\star(\mathbf{z}_2)$.*

Proof. For a contradition, suppose not, so that $\mathcal{J}^o \equiv \mathcal{J}^\star(\mathbf{z}_1) \setminus \mathcal{J}^\star(\mathbf{z}_2) \neq \emptyset$. Let $\mathcal{J}^i \equiv \mathcal{J}^\star(\mathbf{z}_1) \cap \mathcal{J}^\star(\mathbf{z}_2)$ so that $\mathcal{J}^\star(\mathbf{z}_1) = \mathcal{J}^i \cup \mathcal{J}^o$. Observe that, for any convex combination of the two types $\mathbf{z}_1 + \theta(\mathbf{z}_2 - \mathbf{z}_1)$, the marginal value of item j in decision set $\mathcal{J}$ for this type can be expressed using the line integral

$$D_j\pi(\mathcal{J}; \mathbf{z}_1 + \theta(\mathbf{z}_2 - \mathbf{z}_1)) = D_j\pi(\mathcal{J}; \mathbf{z}_1) + \int_0^\theta \nabla D_j\pi(\mathcal{J}; \mathbf{z}_1 + t(\mathbf{z}_2 - \mathbf{z}_1)) \cdot (\mathbf{z}_2 - \mathbf{z}_1)\mathrm{d}t\,.$$

Since the integrand is positive for $\theta > 0$, the integral is positive as well. We may conclude that type $\mathbf{z}_1 + \theta(\mathbf{z}_2 - \mathbf{z}_1)$ receives higher marginal benefit from j's addition in $\mathcal{J}$ than type $\mathbf{z}_1$.

Claim: agents of type $\mathbf{z}_2$ prefer $\mathcal{J}^\star(\mathbf{z}_2) \cup \mathcal{J}^o$ to $\mathcal{J}^\star(\mathbf{z}_2)$, a contradiction since $\mathcal{J}^o$ is non-empty. We prove this claim by induction on n the cardinality of J^o. To begin, suppose $|J^o| = 1$ and let its element j. Then,

$$\begin{aligned} 0 &\leq D_j\pi(\mathcal{J}^\star(\mathbf{z}_1), \mathbf{z}_1) \\ &= \pi(\mathcal{J}^i \cup \{j\}; \mathbf{z}_1) - \pi(\mathcal{J}^i, \mathbf{z}_1) \\ &\leq \pi(\mathcal{J}^i \cup \{j\}, \mathbf{z}_2) - \pi(\mathcal{J}^i; \mathbf{z}_2) \\ &= D_j(\mathcal{J}^i; \mathbf{z}_2) \end{aligned}$$

where the inequality follows from the line integral above. Then, $0 \geq D_j(\mathcal{J}^{i}\mathbf{z}_2)$, implying that $0 \leq D_j(\mathcal{J}^\star(\mathbf{z}_2); \mathbf{z}_2)$ from SCD-C from below. Then, j should optimally be included and $\mathcal{J}^\star(\mathbf{z}_2) \cup \mathcal{J}^o$ is preferred to $\mathcal{J}^\star(\mathbf{z}_2)$. The claim holds for $n = 1$.

Suppose the claim holds for n. To show it must hold for $n+1$, suppose $|J^o| = n+1$ and select an item from this set to label j. Let $\mathcal{J}^o_n = \mathcal{J}^o \setminus \{j\}$. Then,

$$\begin{aligned}
0 &\leq D_j\pi(\mathcal{J}^\star(\mathbf{z}_1), \mathbf{z}_1) = D_j\pi(\mathcal{J}^i \cup \mathcal{J}^o_n, \mathbf{z}_1) \\
&\leq D_j\pi(\mathcal{J}^i \cup \mathcal{J}^o_n, \mathbf{z}_2) \\
0 &\leq D_j\pi(\mathcal{J}^\star(\mathbf{z}_2) \cup \mathcal{J}^o_n, \mathbf{z}_2)
\end{aligned}$$

where the second line follows from the line integral and the third from SCD-C from below. We may conclude that an agent with type $\mathbf{z}_2$ prefers $\mathcal{J}^\star(\mathbf{z}_2) \cup \mathcal{J}^o_n \cup \{j\}$ to $\mathcal{J}^\star(\mathbf{z}_2) \cup \mathcal{J}^o_n$, which is itself preferred to $\mathcal{J}^\star(\mathbf{z}_2)$ by the inductive assumption.

We have shown that, for any finite non-empty $\mathcal{J}^o$, agents of type $\mathbf{z}_2$ prefer $\mathcal{J}^\star(\mathbf{z}_2) \cup \mathcal{J}^o$ to $\mathcal{J}^\star(\mathbf{z}_2)$, a contradiction. $\square$

In particular, observe that the condition replacing SCD-T is stricter than the sufficiency condition for SCD-T provided above. The sufficiency condition allows the ith component of the gradient to differ for each i, j, and $\mathcal{J}$, as long as it is the same across the typespace given these. On the other hand, the condition provided for the multidimensional nesting result requires the sign to remain the same for each i, regardless of which j and $\mathcal{J}$ are chosen.

C. Input sourcing and value chain framework

We first show that the unit cost of production in our GVC framework is proportional to $\Theta_m(\{\mathcal{J}^k\}_k)^{-1/\theta}$. We then show that an alternative framework where each intermediate is a "snake" yields the same firm decisions and allocations as our baseline framework.

Begin by considering each destination market separately since the firm's activities in one do not affect the firm's outcomes in another. Fix a destination market m and proceed by induction on the number of stages K. Begin first with $K = 2$ and consider the unit cost implied by a given sourcing strategy $\{j^1(\omega^1), j^2(\omega^2)\}$. The first stage is produced with labor only, so the cost of producing any intermediate ω^1 is $w_{j^1(\omega^1)}$. Then, the total unit cost conditional on the sourcing strategy is

$$\exp\left\{\int\int \ln\left[\frac{d_{j^2(\omega^2)m}}{\nu(\omega^2)}\left(w_{j^1(\omega^1)}d_{j^1(\omega^1)j^2(\omega^2)}\right)^{\alpha^2} w_{j^2(\omega^2)}^{1-\alpha^2}\right] \mathrm{d}\omega^1 \mathrm{d}\omega^2\right\}$$

where we have used the Cobb-Douglas structure of production.

For the inductive step, we assert that unit costs given a sourcing strategy are

$$\exp\left\{\int \ln\left[\frac{1}{\nu(\omega^K)}\prod_{k=1}^{K} w_{j^k(\omega^k)}^{\beta^k} d_{j^k(\omega^k)j^{k+1}(\omega^{k+1})}^{\frac{\alpha^{k+1}}{1-\alpha^{k+1}}\beta^{k+1}}\right] \mathrm{d}\omega^1 \ldots \mathrm{d}\omega^K\right\}$$

for a production structure with K stages. We now show the statement holds for production structures with $K+1$ stages. Let this expression omitting the shipping shock be denoted $\tilde{a}_m$ and consider the final stage of production, where a unit continuum of ω^K intermediates must be sourced for each ω^{K+1}. The cost of doing so is $\tilde{a}_{j^{K+1}(\omega^{K+1})}$. Using the properties of Cobb-Douglas production, the final unit cost given a sourcing strategy will therefore be

$$\begin{aligned}
&\exp\left\{\int \ln\left[\frac{d_{j^{K+1}(\omega^{K+1})m}}{\nu(\omega^{K+1})} w_{j^{K+1}(\omega^{K+1})}^{1-\alpha^{K+1}}\left(\tilde{a}_{j^{K+1}(\omega^{K+1})}\right)^{\alpha^{K+1}}\right] \mathrm{d}\omega^{K+1}\right\} \\
=\ &\exp\left\{\int \ln\left[\frac{1}{\nu(\omega^K+1)}\prod_{k=1}^{K+1} w_{j^k(\omega^k)}^{\beta^k} d_{j^k(\omega^k)j^{k+1}(\omega^{k+1})}^{\frac{\alpha^{k+1}}{1-\alpha^{k+1}}\beta^{k+1}}\right] \mathrm{d}\omega^1 \ldots \mathrm{d}\omega^K\right\}
\end{aligned}$$

completing the inductive step.

What remains is to characterize the unit costs of the firm once it has selected the optimal sourcing strategy. Since there is a continuum of final-stage intermediates ω^K, the Fréchet distribution of shipping shocks $\nu(\omega^K)$ is exactly realized at the firm level. Thus, for each intermediate ω^K, a shipping strategy is chosen conditional on the shock draws

$$\begin{aligned}
a_m(\{\mathcal{J}^k\}_k) &= \exp\left\{\int \ln\left[\min_{\{j^k(\omega^k)\}_k} \nu(\omega^K)\prod_{k=1}^{K} w_{j_m^k(\omega^k)}^{\beta^k} d_{j_m^k(\omega^k)j_m^{k+1}(\omega^{k+1})}^{\frac{\alpha^{k+1}}{1-\alpha^{k+1}}\beta^{k+1}}\right] \mathrm{d}\omega^1 \ldots \mathrm{d}\omega^K\right\} \\
a_m(\{\mathcal{J}^k\}_k) &\equiv \exp\left\{\mathbb{E}\left[\ln M\right]\right\} \\
&= \exp\left\{\ln \Theta_m(\{\mathcal{J}^k\}_k)^{-\frac{1}{\theta}} - \frac{\gamma}{\theta}\right\} \propto \Theta_m(\{\mathcal{J}^k\}_k)^{-\frac{1}{\theta}}
\end{aligned}$$

where M denotes the minimum. This minimum is distributed Weilbull with shape θ and scale $\Theta_m(\{\mathcal{J}^k\}_k)^{-1/\theta}$, given the shocks are Fréchet distributed. The third line follows from the properties of the Weilbull distribution, where γ is the Euler–Mascheroni constant.

We now explore an alternative framework where each ω^K intermediate is instead produced with a snake-structure, so that each ω^K corresponds to the path of an initial intermediate ω^1 being progressively transformed as it moves through the value chain. In particular, suppose that the previous stage input is combined Cobb-Douglas with labor each stage. The first stage remains produced with labor only. The firm's total production is then

$$q_m = \exp\left\{ \int \ln\left[\frac{\nu^K(\omega^K)}{d_{j^K(\omega^K)m}} q^K(\omega^K) \right] \mathrm{d}\omega^K \right\} \qquad q^k(\omega^K) = \left(\frac{q^{k-1}(\omega^K)}{d_{j^{k-1}(\omega^K)j^k(\omega^K)}\alpha^k} \right)^{\alpha^k} \left(\frac{\ell^k(\omega^k)}{1-\alpha^k} \right)^{1-\alpha^k}$$

given a path $\{j^k(\omega^K)\}_j$ for each intermediate ω^K. Through a similar inductive argument to above, the unit cost of production given a set of paths is

$$\exp\left\{ \int \ln\left[\frac{1}{\nu(\omega^K)} \prod_{k=1}^{K} w^{\beta^k}_{j^k(\omega^K)} d^{\frac{\alpha^{k+1}}{1-\alpha^{k+1}}\beta^{k+1}}_{j^k(\omega^K)j^{k+1}(\omega^K)} \right] \mathrm{d}\omega^K \right\}$$

where the β^ks once again represent the overall weight on labor from each stage. We now assume the shipping shocks $\nu(\omega^K)$ for an intermediate ω^K are drawn for each potential path, with the firm choosing the path yielding the lowest cost. Once again appealing to the law of large numbers, the Fréchet distribution of shocks is realized across the continuum of intermediates ω^K. In a similar argument to above, the unit cost of production is therefore identical to the one derived above. Moreover, on aggregate, each location-stage $j \in \mathcal{J}^k$ carries out the same amount of production in both frameworks. This share is encapsulated described by their contribution to the network potential term, as is common in gravity models with a multilateral access term.