Nonstationary Dynamic Models with Finite Dependence∗

Peter Arcidiacono Robert A. Miller
Duke University & NBER Carnegie Mellon University

October 14, 2015

Abstract

The estimation of non-stationary dynamic discrete choice models typically requires making assumptions far beyond the length of the data. We extend the class of dynamic discrete choice models that require only a few-period-ahead conditional choice probabilities as well as developing algorithms to calculate the finite dependence paths. We do this both in single agent and games settings, resulting in expressions for the value functions that allow for much weaker assumptions regarding the time horizon and the transitions of the state variables beyond the sample period.

1 Introduction

Estimation of dynamic discrete choice models is complicated by the calculation of expected future payoffs. These complications are particularly pronounced in games where the equilibrium actions and future states of the other players must be margined out to derive a player’s best response. Originating with Hotz and Miller (1993), two-step methods provide a way of cheaply estimating structural payoff parameters in both single-agent and multi-agent settings. These two-step estimators first estimate

∗We thank Victor Aguirregabiria, Shakeeb Khan, Jean-Marc Robin, and seminar participants at Duke, Sciences Po, Toulouse, and Toronto for helpful comments. We acknowledge support from National Science Foundation Grant Awards SES0721059 and SES0721098.
conditional choice probabilities (CCP’s) and then characterize a portion of the future payoffs as function of the CCP’s when estimating the structural payoff parameters.¹

CCP estimators fall into two classes: those that exploit finite dependence, and those that do not.² The former entails expressing the future value term or its difference across two alternatives as a function of just a few-period ahead conditional choice probabilities and flow payoffs.³ Intuitively, ρ period finite dependence holds in a dynamic discrete choice optimization model when there exist two (potentially weighted) sequences of choices, neither of which is necessarily optimal for the player, that lead off from different initial choices but, through successive state variable transitions, generate the same distribution of state variables ρ + 1 periods later.

The advantage of working with a finite dependence representation is that the stationarity assumption can be relaxed along with assumptions about the length of the time horizon and the evolution of the state variables well beyond the periods covered in the data. For example, a dynamic model of schooling requires making assumptions regarding the age of retirement but the data available to researchers may only track individuals into their twenties or thirties. Further, since only a few-period-ahead conditional choice probabilities are needed, computational times are also reduced.

Many papers have used the finite dependence property in estimation, often employing either a terminal or renewal action.⁴ More general forms of finite dependence, whether a feature of the data

¹ See Arcidiacono and Ellickson (2011) for a review.
⁴ See, for example Hotz and Miller (1993), Joensen (2009), Scott (2013), Arcidiacono, Bayer, Blevins, and Ellickson (2013), Decker and Verboven (2014), Mazur (2014), and Beauchamp (2015). The last three exploit one period finite dependence to estimate dynamic games.
or imposed by the authors, have been applied in models of migration (Bishop, 2012, Coate 2013, Ma 2013, Ransom 2014), smoking (Matsumoto 2014), education (Arcidiacono, Aucejo, Maurel, and Ransom 2014), occupational choice (James 2014), fertility and female labor supply (Altug and Miller 1998, Gayle and Golan 2012, Gayle and Miller 2014), housing choices (Khorunzhina and Miller 2014), participation in the stock market (Khorunzhina 2013), and agricultural land use (Scott, 2013). These papers demonstrate the advantage of exploiting finite dependence in estimation: it is not necessary to solve the value function within a nested fixed point algorithm, nor invert matrices the size of the state space.\footnote{The finite dependence property has also been directly imposed on the decision making process in models to economize on the state space. See for example Bishop (2012) and Ma (2013). By assuming agents do not use all relevant information agents have at their disposal, the state space that agents use to solve their optimization problems can be reduced. This approach provides a parsimonious way of modeling bounded rationality when the state space is high dimensional.}

Our approach is linear, even when parallel processing is used to check for finite dependence. Moreover many of the resulting estimators have an intrinsically linear structure. From the standpoint of computational efficiency, the advantages of linear solution methods over nonlinear ones are well known. The Monte Carlo applications given in our previous work (Arcidiacono and Miller, 2011) compare CCP estimators exploiting the finite dependence and linearity with nonlinear Maximum Likelihood estimator. We find the CCP estimators are much cheaper to compute and are almost as precise even in low dimensional problems, where nonlinear methods are least likely to be computationally burdensome.

The current method for determining whether finite dependence holds or not is to guess and verify. The main contribution of this paper is provide a systematic way of determining whether finite dependence holds. To accomplish this, we slightly generalize the definition of finite dependence given in Arcidiacono and Miller (2011), which in turn extends the class of models that can be cheaply
estimated. Key to the generalization is recognizing that the ex-ante value function can be expressed as a weighted average of the conditional value functions of all the alternatives plus a function of the conditional choice probabilities, where the weights sum to one, but do not all need to be positive. This turns the search for finite dependence into a straightforward algorithm: successively check for a nonzero determinant in order to eventually solve a linear system of equations that produces the finite dependence weights. The linear system generally has a low dimension, because it is based on the number of states that are attainable in a few periods from the initial state, not the size of the state space itself.

In game settings, finite dependence is applicable to each player individually. Here finite dependence relates to transition matrices for the state variables when a designated player follows arbitrary mixed strategies (that might include, following our extended definition, counter-intuitive negative weights) and the other players follow their equilibrium strategies. Consequently, finite dependence in games cannot be ascertained from the transition primitives alone (as in the individual optimization case). Indeed, whether or not finite dependence holds, might also hinge on which equilibrium is played, not a paradoxical result, because different equilibrium for the same game sometimes reveal different information about the primitives, so naturally require different estimation approaches.

Up until now research on finite dependence in games has been restricted to cases where there is a terminal or renewal action (that ends or restarts the process governing the state variables for individual players). Absent these two cases, one-period finite dependence fails to hold, because the equilibrium actions of the other players depend on what the designated agent has already done, and hence the distribution of the state variables, which they partly determine, depends on the actions of the designated player two periods earlier. These stochastic connections, a vital feature of many strategic interactions, has limited empirical research in estimating games with nonstationarities. We develop an algorithm to solve for finite dependence in a broader class of games than those
characterized by terminal and renewal actions. As in the single agent case, the algorithm entails solving a linear system of equations where the number of equations is dictated by the possible states that can be reached a few periods ahead.

The rest of the paper proceed as follows. Section 2 lays out our framework for analyzing finite dependence in discrete choice dynamic optimization and games. In Section 3 we define finite dependence, provide a new representation of this property, and use the representation to demonstrate how to recover finite dependence paths in both single agent and multi-agent settings. New examples of finite dependence, derived using the algorithm, are provided in Section 4, while Section 5 concludes with some remarks on outstanding questions that future research might address.

2 Framework

This section first lays out a general class of dynamic discrete choice models. Drawing upon our previous work (Arcidiacono and Miller, 2011), we extend our representation of the conditional value functions which plays an overarching role in our analysis, and then modify our framework to accommodate games with private information.

2.1 Dynamic optimization discrete choice

In each period \( t \in \{1, \ldots, T\} \) until \( T \leq \infty \), an individual chooses among \( J \) mutually exclusive actions. Let \( d_{jt} \) equal one if action \( j \in \{1, \ldots, J\} \) is taken at time \( t \) and zero otherwise. The current period payoff for action \( j \) at time \( t \) depends on the state \( x_t \in \{1, \ldots, X\} \).\(^6\) If action \( j \) is taken at time \( t \), the probability of \( x_{t+1} \) occurring in period \( t + 1 \) is denoted by \( f_{jt}(x_{t+1}|x_t) \).

The individual’s current period payoff from choosing \( j \) at time \( t \) is also affected by a choice-
specific shock, \( \epsilon_{jt} \), which is revealed to the individual at the beginning of the period \( t \). We assume the vector \( \epsilon_t \equiv (\epsilon_{1t}, \ldots, \epsilon_{Jt}) \) has continuous support, is drawn from a probability distribution that is independently and identically distributed over time with density function \( g(\epsilon_t) \), and satisfies \( E[\max \{\epsilon_{1t}, \ldots, \epsilon_{Jt}\}] \leq M < \infty \). The individual’s current period payoff for action \( j \) at time \( t \) is modeled as \( u_{jt}(x_t) + \epsilon_{jt} \).

The individual takes into account both the current period payoff as well as how his decision today will affect the future. Denoting the discount factor by \( \beta \in (0, 1) \), the individual chooses the vector \( d_t \equiv (d_{1t}, \ldots, d_{Jt}) \) to sequentially maximize the discounted sum of payoffs:

\[
E \left\{ \sum_{t=1}^{T} \sum_{j=1}^{J} \beta^{t-1} d_{jt} \left[ u_{jt}(x_t) + \epsilon_{jt} \right] \right\} \tag{1}
\]

where at each period \( t \) the expectation is taken over the future values of \( x_{t+1}, \ldots, x_T \) and \( \epsilon_{t+1}, \ldots, \epsilon_T \).

Expression (1) is maximized by a Markov decision rule which gives the optimal action conditional on \( t, x_t, \) and \( \epsilon_t \). We denote the optimal decision rule at \( t \) as \( d^o_{jt}(x_t, \epsilon_t) \), with \( j^{th} \) element \( d^o_{jt}(x_t, \epsilon_t) \).

The probability of choosing \( j \) at time \( t \) conditional on \( x_t, p_{jt}(x_t) \), is found by taking \( d^o_{jt}(x_t, \epsilon_t) \) and integrating over \( \epsilon_t \):

\[
p_{jt}(x_t) \equiv \int d^o_{jt}(x_t, \epsilon_t) g(\epsilon_t) \, d\epsilon_t \tag{2}
\]

We then define \( p_t(x_t) \equiv (p_{1t}(x_t), \ldots, p_{Jt}(x_t)) \) as the vector of conditional choice probabilities.

Denote \( V_t(x_t) \), the ex-ante value function in period \( t \), as the discounted sum of expected future payoffs just before \( \epsilon_t \) is revealed and conditional on behaving according to the optimal decision rule:

\[
V_t(x_t) \equiv E \left\{ \sum_{\tau=t}^{T} \sum_{j=1}^{J} \beta^{\tau-t} d^o_{jt}(x_\tau, \epsilon_\tau) \left[ u_{j\tau}(x_\tau) + \epsilon_{j\tau} \right] \right\}
\]

Given state variables \( x_t \) and choice \( j \) in period \( t \), the expected value function in period \( t+1 \), discounted one period into the future is \( \beta \sum_{x_{t+1}=1}^{X} V_{t+1}(x_{t+1}) f_{jt}(x_{t+1}|x_t) \). Under standard conditions, Bellman’s principle applies and \( V_t(x_t) \) can be recursively expressed as:

\[
V_t(x_t) = \sum_{j=1}^{J} \int d^o_{jt}(x_t, \epsilon_t) \left[ u_{jt}(x_t) + \epsilon_{jt} + \beta \sum_{x_{t+1}=1}^{X} V_{t+1}(x_{t+1}) f_{jt}(x_{t+1}|x_t) \right] g(\epsilon_t) \, d\epsilon_t
\]
We then define the choice-specific conditional value function, \( v_{jt}(x_t) \), as the flow payoff of action \( j \) without \( \epsilon_{jt} \) plus the expected future utility conditional on following the optimal decision rule from period \( t + 1 \) on:\(^7\)

\[
v_{jt}(x_t) = u_{jt}(x_t) + \beta \sum_{x_{t+1}=1}^{X} V_{t+1}(x_{t+1}) f_{jt}(x_{t+1}|x_t)
\]

Our analysis is based on a representation of \( v_{jt}(x_t) \) that slightly generalizes Theorem 1 of Arcidiacono and Miller (2011). Both results are based on their Lemma 1, that for every \( t \in \{1, \ldots, T\} \) and \( p \in \Delta^J \), the \( J \) dimensional simplex, there exists a real-valued function \( \psi_j(p) \) such that:

\[
\psi_j[p_t(x)] \equiv V_t(x) - v_{jt}(x)
\] \(^{(4)}\)

To interpret (4), note that the value of committing to action \( j \) at period \( t \) before seeing \( \epsilon_t \) and behaving optimally thereafter is \( v_{jt}(x_t) + E[\epsilon_{jt}] \). Therefore the expected loss from pre-committing to \( j \) versus waiting until \( \epsilon_t \) is observed and only then making an optimal choice, \( V_t(x_t) \), is the constant \( \psi_j[p_t(x_t)] \) minus \( E[\epsilon_{jt}] \), a composite function that only depends \( x_t \) through the conditional choice probabilities. This result leads to the following theorem, proved using an induction.

**Theorem 1** For each choice \( j \in \{1, \ldots, J\} \) and \( \tau \in \{t + 1, \ldots, T\} \), let any \( \omega_{\tau}(x_{\tau}, j) \) denote any mapping from the state space \( \{1, \ldots, X\} \) to \( R^J \) satisfying the constraints that \( |\omega_{k\tau}(x_{\tau}, j)| \leq B \) for some \( B < \infty \) and \( \sum_{k=1}^{J} \omega_{k\tau}(x_{\tau}, j) = 1 \). Recursively define \( \kappa_{\tau}(x_{\tau+1}|x_t, j) \) as:

\[
\kappa_{\tau}(x_{\tau+1}|x_t, j) \equiv \begin{cases} f_{jt}(x_{t+1}|x_t) & \text{for } \tau = t \\ \sum_{x_{\tau}=1}^{X} \sum_{k=1}^{J} \omega_{k\tau}(x_{\tau}, j) f_{k\tau}(x_{\tau+1}|x_{\tau}) \kappa_{\tau-1}(x_{\tau}|x_t, j) & \text{for } \tau = t + 1, \ldots, T \end{cases}
\] \(^{(5)}\)

Then for \( T < T \):

\[
v_{jt}(x_t) = u_{jt}(x_t) + \sum_{\tau=t+1}^{T} \sum_{k=1}^{J} \sum_{x_{\tau}=1}^{X} \beta^{\tau-t} [u_{k\tau}(x_{\tau}) + \psi_k[p_\tau(x_{\tau})]] \omega_{k\tau}(x_{\tau}, j) \kappa_{\tau-1}(x_{\tau}|x_t, j) \]

\[
+ \sum_{x_{\tau+1}}^{X} \beta^{\tau+1-t} V_{\tau+1}(x_{\tau+1}) \kappa_{\tau}(x_{\tau+1}|x_t, j)
\] \(^{(6)}\)

\(^7\)For ease of exposition we refer to \( v_{jt}(x_t) \) as the conditional value function in the remainder of the paper.
and for $\mathcal{T} = T$:

$$v_{jt}(x_t) = u_{jt}(x_t) + \sum_{\tau=t+1}^{T} \sum_{k=1}^{J} \sum_{x_t=1}^{X} \beta^{\tau-t} [u_{k\tau}(x_\tau) + \psi_k[p_{\tau}(x_\tau)] \omega_{k\tau}(x_\tau, j) \kappa_{\tau-1}(x_\tau|x_t, j)$$

(7)

Arcidiacono and Miller (2011) prove the theorem when $T = \mathcal{T}$ and $\omega_{k\tau}(x_\tau, j) \geq 0$ for all $k$ and $\tau$.

In that case, $\kappa_{\tau}(x_{\tau+1}|x_t, j)$ is the probability of reaching $x_{\tau+1}$ by following the sequence defined by $\omega_{\tau}(x_\tau, j)$ and the value function representation extending over the whole decision-making horizon.$^8$

### 2.2 Extension to dynamic games

This framework extends naturally to dynamic games. In the games setting, we assume that there are $N$ players making choices in periods $t \in \{1, \ldots, T\}$. The systematic part of payoffs to the $n^{th}$ player not only depends on his own choice in period $t$, denoted by $d_t^{(n)} \equiv (d_t^{(n)}_1, \ldots, d_t^{(n)}_J)$, the state variables $x_t$, but also the choices of the other players, which we now denote by $d_t^{(\sim n)} \equiv (d_t^{(1)}, \ldots, d_t^{(n-1)}, d_t^{(n+1)}, \ldots, d_t^{(N)})$. Denote by $U_{jt}^{(n)}(x_t, d_t^{(\sim n)}) + \epsilon_{jt}^{(n)}$ the current utility of player $n$ in period $t$, where $\epsilon_{jt}^{(n)}$ is an identically and independently distributed random variable that is private information to the firm. Although the players all face the same observed state variables, these state variables typically affect players in different ways. For example, adding to the $n^{th}$ player’s capital may increase his payoffs and reduce the payoffs to the others. For this reason the payoff function is superscripted by $n$.

Each period the players make simultaneous choices. We denote by $P_t\left(d_t^{(\sim n)} | x_t\right)$ the joint conditional choice probability that the players aside from $n$ collectively choose $d_t^{(\sim n)}$ at time $t$ given the state variables $x_t$. Since $\epsilon_t^{(n)}$ is independently distributed across all the players, $P_t\left(d_t^{(\sim n)} | x_t\right)$ has the product representation:

$$P_t\left(d_t^{(\sim n)} | x_t\right) = \prod_{n'=1}^{I} \left( \sum_{j=1}^{J} \sum_{n'} d_j^{(n')} p_j^{(n')} (x_t) \right)$$

(8)

$^8$The extension to negative weights is also noted in Gayle (2013).
We assume each player acts like a Bayesian when forming his beliefs about the choices of the other players and that a Markov-perfect equilibrium is played. Hence, the beliefs of the players match the probabilities given in equation (8). Taking the expectation of $U_{jt}^{(n)}(x_t, d_t^{(n)})$ over $d_t^{(n)}$, we define the systematic component of the current utility of player $n$ as a function of the state variables as:

$$u_{jt}^{(n)}(x_t) = \sum_{d_t^{(n)} \in J^{N-1}} P_t(d_t^{(n)} | x_t) U_{jt}^{(n)}(x_t, d_t^{(n)})$$

(9)

For future reference we call $u_{jt}^{(n)}(x_t)$ the reduced form payoff to player $n$ from taking action $j$ in period $t$ when the state is $x_t$.

The values of the state variables at period $t + 1$ are determined by the period $t$ choices by all the players as well as the values of the period $t$ state variables. We consider a model in which the state variables can be partitioned into those that are affected by only one of the players, and those that are exogenous. For example, to explain the number and size of firms in an industry, the state variables for the model might be indicators of whether each potential firm is active or not, and a scalar to measure firm capital or capacity; each firm controls their own state variables, through their entry and exit choices, as well as their investment decisions.\(^9\) The partition can be expressed as $x_t \equiv (x_t^{(0)}, x_t^{(1)}, \ldots, x_t^{(N)})$, where $x_t^{(0)}$ denotes the states that are exogenously determined by transition probability $f_0(x_t^{(0)} | x_t^{(0)})$, and $x_t^{(n)} \in X^{(n)} \equiv \{1, \ldots, X^{(n)}\}$ is the component of the state controlled or influenced by player $n$. Let $f_{jt}^{(n)}(x_{t+1}^{(n)} | x_t^{(n)})$ denote the probability that $x_{t+1}^{(n)}$ occurs at time $t + 1$ when player $n$ chooses $j$ at time $t$ given $x_t^{(n)}$. Many models in industrial organization exploit this specialized structure because it provides a flexible way for players to interact while keeping the model simple enough to be empirically tractable.\(^10\)

---

\(^9\)The second example in Arcidiacono and Miller (2011) also belongs to this class of models.

\(^10\)All the empirical applications of structural modeling of which we are aware have this property including, for example, those based Ericson and Pakes (1995). Namely, firms affect their own product quality through their choices of investment decisions but do not directly affect the product quality of other players. The firm’s decisions affect product quality of other players only through their effect on the decisions of the other players.
Denote the state variables associated with all the players aside from \( n \) as:

\[
x_t^{(\sim n)} = \left( x_t^{(1)}, \ldots, x_t^{(n-1)}, x_t^{(n+1)} \ldots, x_t^{(N)} \right) \in \mathcal{X}(\sim n) \equiv \mathcal{X}(1) \times \ldots \times \mathcal{X}(n-1) \times \mathcal{X}(n+1) \times \ldots \times \mathcal{X}(N)
\]

Under this specification the reduced form transition generated by their equilibrium choice probabilities is defined as:

\[
f_t^{(\sim n)}(x_{t+1} | x_t) = \prod_{n' = 1}^{N} \left[ \sum_{k=1}^{J} \rho_{kt}^{(n')} (x_t) f_{kt}^{(n')} \left( x_{t+1} | x_t^{(n')} \right) \right]
\]

As in Subsection 2.1, consider for all \( \tau \in \{ t, \ldots, T \} \) any sequence of decision weights:

\[
\omega^{(n)}_{\tau}(x_\tau, j) \equiv \left( \omega^{(n)}_{1\tau}(x_\tau, j), \ldots, \omega^{(n)}_{J\tau}(x_\tau, j) \right)
\]

subject to the constraints \( \sum_{k=1}^{J} \omega_{k\tau}(n)(x_\tau, j) = 1 \) and starting value \( \omega_{j\tau}^{(n)}(x_t, j) = 1 \). Given the equilibrium actions of the other players impounded in \( f_t^{(\sim n)}(x_{t+1} | x_t) \), we recursively define \( \kappa^{(n)}_{\tau}(x_{\tau+1} | x_t, j) \) for the sequence of decision weights \( \omega^{(n)}_{k\tau}(x_\tau, j) \) over periods \( \tau \in \{ t + 1, \ldots, T \} \) in a similar manner to Equation (5) as:

\[
\kappa^{(n)}_{\tau}(x_{\tau+1} | x_t, j) \equiv f_{00} \left( x_{\tau+1}^{(0)} | x_\tau^{(0)} \right) \sum_{x_{\tau+1}^{(0)}} f^{(n)}_{x_{\tau+1} | x_\tau} \sum_{k=1}^{J} \omega^{(n)}_{k\tau}(x_\tau, j) f^{(n)}_{x_{\tau+1}^{(n)} | x_\tau^{(n)}} \kappa^{(n)}_{\tau-1}(x_\tau, x_t, j)
\]

with initializing function:

\[
\kappa^{(n)}_{t}(x_{t+1} | x_t, j) \equiv f_{jt}^{(n)} \left( x_{t+1}^{(n)} | x_t^{(n)} \right) f_t \left( x_{t+1}^{(n)} | x_t^{(n)} \right) f_{00} \left( x_{t+1}^{(0)} | x_t^{(0)} \right)
\]

Letting:

\[
f_{jt} (x_{t+1} | x_t) = f_{00} \left( x_{t+1}^{(0)} | x_t^{(0)} \right) f_t^{(n)} \left( x_{t+1}^{(n)} | x_t^{(n)} \right) f_{jt}^{(n)} \left( x_{t+1}^{(n)} | x_t^{(n)} \right)
\]

and adding \( n \) superscripts to all the other terms in (7), it now follows that Theorem 1 applies to this multi-agent setting in exactly the same way as in a single agent setting.

## 3 Finite dependence

If there were transition matrices satisfying the equality \( \kappa_T^{(n)}(x_{\tau+1} | x_t, 1) = \kappa_T^{(n)}(x_{\tau+1} | x_t, j) \), then (6) implies differences in the conditional value functions \( v_{jt}(x_t) - v_{1t}(x_t) \) could be expressed as weighted
sums of the parameters determining utilities, along with $\psi_k(\cdot)$ known functions of the identified CCP’s, that occur between $t$ and $T$, the end of the sample, rather than $t$ and $T$, the end of the individual’s problem solving horizon. Finite dependence is the natural generalization of an equality like $\kappa_T^*(x_{T+1}|x_t, 1) = \kappa_T^*(x_{T+1}|x_t, j)$. It captures the notion that the differential effects on the state variable from taking two distinct actions in period $t$ might be obliterated, say $\rho$ periods later, if certain corrective paths are followed that are specific to the initial action.

3.1 Defining finite dependence

Consider two sequences of decision weights that begin at date $t$ in state $x_t$, one with choice $i$ and the other with choice $j$. We say that the pair of choices $\{i, j\}$ exhibits $\rho$-period dependence if there exists sequences of decision weights from $i$ and $j$ such that:

$$\kappa_{t+\rho}(x_{t+\rho+1}|x_t, i) = \kappa_{t+\rho}(x_{t+\rho+1}|x_t, j) \quad (10)$$

for all $x_{t+\rho+1}$. That is, the weights associated with each state are the same across the two paths after $\rho$ periods.\(^{11}\) Trivially, finite dependence holds in all finite horizon problems. However the property of $\rho$-period dependence only merits attention when $\rho < T - t$. To avoid repeatedly referencing the trivial case of $\rho = T - t$, we will henceforth write finite dependence holds only when (10) applies for $\rho < T - t$.

Under finite dependence, differences in current utility $u_{jt}(x_t) - u_{it}(x_t)$ can be expressed as:

$$u_{jt}(x_t) - u_{it}(x_t) = \psi_i[p_t(x_t)] - \psi_j[p_t(x_t)]$$

$$+ \sum_{\tau=t+1}^{t+\rho} \sum_{k=1}^{J} \sum_{x_\tau=1}^{X} \beta^{\tau-t} \left\{ u_{k\tau}(x_\tau) + \psi_k[p_\tau(x_\tau)] \right\} \begin{bmatrix} \omega_{k\tau}(x_\tau, i) \kappa_{\tau-1}(x_\tau|x_t, i) \\ -\omega_{k\tau}(x_\tau, j) \kappa_{\tau-1}(x_\tau|x_t, j) \end{bmatrix}$$

\(^{11}\)Aguirregabiria and Magaesan (2013) and Gayle (2013) restrict their analyses to cases where there is one period finite dependence, thus ruling out labor supply applications such as Altug and Miller (1998) and games that do not have a terminal choice.
This equation follows directly from Equations (4) and (7), in Theorem 1.\textsuperscript{12} As the empirical applications of finite dependence illustrate, Equations like (11) provide the basis for estimation without resorting to the inversion of high dimension matrices or long simulations. Aside from its computational benefits, finite dependence has a second attractive feature, empirical content, because it is straightforward to test whether (7) is rejected by the data.

\subsection*{3.2 One-period finite dependence in optimization problems}

Because the guess and verify method is essentially the only method researchers have to used to determine finite dependence, it is not surprising that almost all empirical applications of finite dependence have exploited two special cases of one-period finite dependence, models with terminal or renewal choices. Terminal choices end the optimization problem or game by preventing any future decisions; irreversible sterilization against future fertility, (Hotz and Miller, 1993) and firm exit from an industry (Aguirregabiria and Mira, 2007; Pakes, Ostrovsky, and Berry, 2007) are examples. The defining feature of a renewal choice is that it resets the states that were influenced by past actions. Turnover and job matching (Miller, 1984), or replacing a bus engine (Rust, 1987), are illustrative of renewal actions. Let the first choice denote the terminal or renewal choice. In such models, following any choice \( j \in \{1, \ldots, J\} \) with a terminal or renewal choice leads to same value of state variables after two periods. Thus for all \( t < T \) and \( x_t \) the probability distribution of \( x_{t+2} \) conditional on \( x_t \) does not depend on the choice made in period \( t \) if the terminal or renewal choice is taken in period \( t + 1 \):

\[
\sum_{x_{t+1}=1}^{X} f_{1,t+1}(x_{t+2}|x_{t+1})f_{jt}(x_{t+1}|x_t) = \sum_{x_{t+1}=1}^{X} f_{1,t+1}(x_{t+2}|x_{t+1})f_{1t}(x_{t+1}|x_t)
\]  

\textsuperscript{12}Appealing to (4), replace \( v_{jt}(x) \) with \( V_t(x) - \psi_j\left[p_t(x)\right] \) in (7) and perform a similar substitution for \( v_{jt}(x) \). Differencing the two equations results in the terms involving \( V_t(x) \) dropping out.
The difference in conditional value functions between \( j \) and the renewal action, which forms the basis for the estimation of the structural parameters, can then be expressed as:

\[
v_{jt}(x_t) - v_1(x_t) = u_{jt}(x_t) - u_1(x_t) + \beta \sum_{x_{t+1}=1}^X (u_{1t+1}(x_{t+1}) + \psi_1[p_{t+1}(x_{t+1})]) (f_{jt}(x_{t+1}|x_t) - f_{1t}(x_{t+1}|x_t))
\]

(13)

Note that this expression holds regardless of the time horizon, provided the time horizon is greater than or equal to \( t + 1 \). An estimator for parameterizations of \( u_{jt}(x_t) \) can be formed by substituting the right side of (13) for \( v_{jt}(x_t) - v_1(x_t) \) in a discrete choice setup.

But the class of models exhibiting even one-period finite dependence is much larger than terminal and renewal models. From the definition of \( \kappa_{t+1}(x'|x_t, j) \) given by Equation (5), one period finite dependence holds in this specialization if and only if there exists a weighting rule such that \( \kappa_{t+1}(x'|x_t, 1) = \kappa_{t+1}(x'|x_t, 2) \) for all \( x' \in \mathcal{X} \). That is, \( \omega_{2,t+1}(x_{t+1}, j) \) must solve:

\[
\sum_{x_{t+1}=1}^X \left\{ \left[ f_{2,t+1}(x'|x_{t+1}) - f_{1,t+1}(x'|x_{t+1}) \right] \left[ \omega_{2,t+1}(x_{t+1}, 1) f_{1t}(x_{t+1}|x_t) - \omega_{2,t+1}(x_{t+1}, 1) f_{1t}(x_{t+1}|x_t) \right] \right\} = \sum_{x_{t+1}=1}^X f_{1,t+1}(x'|x_{t+1}) \left[ f_{1t}(x_{t+1}|x_t) - f_{2t}(x_{t+1}|x_t) \right]
\]

(14)

for all \( x' \in \mathcal{X} \).

To establish whether or not one period finite dependence exists without relying on the guess and verify method, we now set up the matrix equivalent of (14). It underlies an algorithm that checks for the existence of finite dependence in a finite number of steps. We exploit two features of the system: (i) if any \( x_{t+2} \) cannot be reached regardless of the choices at \( t \) and \( t + 1 \) then the equation associated with that state will automatically be satisfied and (ii) the choices of the weighting rules are only relevant for states at \( t + 1 \) that have positive probabilities of occurring given the initial choice.

Suppose \( A_{j,t+1} \) states can be reached with positive probability in period \( t + 1 \) from state \( x_t \) with choice \( j \) at time \( t \), and denote their set by \( A_{j,t+1} \subseteq \mathcal{X} \). Thus \( x \in A_{j,t+1} \) if and only if \( f_{jt}(x|x_t) > 0 \).
Let $A_{t+2} \subseteq \mathcal{X}$ denote the states that can be reached with positive probability in period $t + 2$ from any element in the union $A_{1,t+1} \cup A_{2,t+1}$ with either action at $t + 1$. Thus $x' \in A_{t+2}$ if and only if $f_{k,t+1}(x'|x) > 0$ for some $x \in A_{1,t+1} \cup A_{2,t+1}$ and $k \in \{1, 2\}$. Finally, denote by $A_{t+2}$ the number of states in $A_{t+2}(x_t)$. It now follows that the matrix-equivalent of Equation (14) reduces to a linear system of $A_{t+2} - 1$ equations with $A_{1,t+1} + A_{2,t+1}$ unknowns.\(^{13}\)

Denote by $K_{jt}(A_{j,t+1})$ the $A_{j,t+1}$ dimensional vector of nonzero probabilities in the string $f_{jt}(1|x_t), \ldots, f_{jt}(X|x_t)$. It gives the one period transition probabilities to $A_{j,t+1}$ from $x_t$ when choice $j$ is made. Let $F_{k,t+1}(A_{j,t+1})$ denote the first $A_{t+2} - 1$ columns\(^{14}\) of the $A_{j,t+1} \times A_{t+2}$ transition matrix from $A_{j,t+1}$ to $A_{t+2}$ when choice $k$ is made in period $t+1$. A typical element of $F_{k,t+1}(A_{j,t+1})$ is $f_{k,t+1}(x'|x)$ where $x \in A_{j,t+1}$ and $x' \in A_{t+2}$. Note that some of the elements of $F_{k,t+1}(A_{j,t+1})$ may be zero. Finally, let $\Omega_{2,t+1}(A_{j,t+1})$ denote an $A_{j,t+1}$ dimensional vector of weights on each of the attainable states at $t+1$ for taking the second choice at that time, comprised of elements $\omega_{2,t+1}(x,j)$ for each $x \in A_{j,t+1}$.

To see how these matrices relate to (14), consider the case when all the states are attainable at both $t + 2$ and $t + 1$ given an initial state $x_t$ and initial choice $k$. The weight associated with each element of $\mathcal{X}$ from taking initial choice $k$ and following that with the weighting rule $\omega_{j,t+1}(x_{t+1}, k)$

---

\(^{13}\)We can remove one equation from the $A_{t+2}$ system because if the weights associated with each state match for $A_{t+2} - 1$ states, they must also match for the remaining state.

\(^{14}\)We focus on the first $A_{t+2} - 1$ columns because the last column must be given by one minus the sum of the previous columns.
is:

\[
\begin{bmatrix}
\sum_{x_{t+1}=1}^X f_{j,t+1}(1|x_{t+1})\omega_{j,t+1}(x_{t+1}, k)f_{kt}(x_{t+1}|x_t) \\
\vdots \\
\sum_{x_{t+1}=1}^X f_{j,t+1}(X|x_{t+1})\omega_{j,t+1}(x_{t+1}, k)f_{kt}(x_{t+1}|x_t)
\end{bmatrix}
\]

\[
= \begin{bmatrix}
f_{j,t+1}(1|1) & \cdots & f_{j,t+1}(1|X) \\
\vdots & \ddots & \vdots \\
f_{j,t+1}(X|1) & \cdots & f_{j,t+1}(X|X)
\end{bmatrix}
\begin{bmatrix}
\omega_{j,t+1}(1, k)f_{kt}(1|x_t) \\
\vdots \\
\omega_{j,t+1}(X, k)f_{kt}(X|x_t)
\end{bmatrix}
\]

\[
\equiv \left[ F_{jt+1}(\mathcal{X}) \right]^\prime \left[ \Omega_{j,t+1}(\mathcal{X}) \circ K_{kt}(\mathcal{X}) \right]
\]

where \( \circ \) refers to element-by-element multiplication. The other terms in (14) are constructed in a similar fashion. When not all the states in \( \mathcal{X} \) are attainable at period \( t+1 \) given choice \( k \), the rows in the second term in the second row of (15) where \( f_{kt}(x_{t+1}|x_t) = 0 \) are removed as well as the corresponding columns of the first term containing the \( f_{j,t+1}(\cdot|x_{t+1}) \) terms. Similarly, if \( x_{t+2} \) is not attainable given either initial choice regardless of the weighting rules at \( t+1 \), then we remove the row of the first term containing the \( f_{j,t+1}(x_{t+2}|\cdot) \) terms. The bottom line of (15) then becomes:

\[
\left[ F_{jt+1}(\mathcal{X}) \right]^\prime \left[ \Omega_{j,t+1}(\mathcal{X}) \circ K_{kt}(\mathcal{X}) \right]
\]

The preceding notation and discussion enables us to express (14) as an \( A_t+2-1 \) system of equations with \( A_{1,t+1} + A_{2,t+1} \) unknowns, namely:\textsuperscript{15}

\[
\begin{bmatrix}
F_{2t+1}(A_{2,t+1}) - F_{1t+1}(A_{2,t+1}) \\
F_{1t+1}(A_{1,t+1}) - F_{2t+1}(A_{1,t+1})
\end{bmatrix}
\]

\[
\left[ \begin{bmatrix}
\Omega_{2,t+1}(A_{2,t+1}) \circ K_{2t}(A_{2,t+1}) \\
\Omega_{2,t+1}(A_{1,t+1}) \circ K_{1t}(A_{1,t+1})
\end{bmatrix}
\right]^\prime 
\begin{bmatrix}
F_{1t+1}(A_{1,t+1}) \\
-F_{1t+1}(A_{2,t+1})
\end{bmatrix}
\]

\[
\left[ \begin{bmatrix}
K_{1t}(A_{1,t+1}) \\
K_{2t}(A_{2,t+1})
\end{bmatrix}
\right]
\]

(16)

If the weights placed on all the other states besides the last are the same across the two paths then the weights placed on the last state must be the same as well. Hence one period finite dependence

\textsuperscript{15}Note that the size of the first matrix on the left (right) hand side, after taking the transpose, is \((A_{t+2} - 1) \times A_{1,t+1} + A_{2,t+1}\), and the size of the second matrix on the left (right) hand side is \((A_{1,t+1} + A_{2,t+1}) \times 1\).
holds if and only if the rank of:

\[
H_{t+1} \equiv \begin{bmatrix}
F_{2t+1}(A_{2,t+1}) - F_{1t+1}(A_{2,t+1}) \\
F_{1t+1}(A_{1,t+1}) - F_{2t+1}(A_{1,t+1})
\end{bmatrix}
\]  \hspace{1cm} (17)

is \(A_{t+2} - 1\).

When \(A_{1,t+1} + A_{2,t+1} = A_{t+2} - 1\), we solve for the weights by inverting (17) and element-by-element dividing both sides of (16) by the relevant \(K\) matrices yielding:

\[
\begin{bmatrix}
\Omega_{2,t+1}(A_{2,t+1}) \\
\Omega_{2,t+1}(A_{1,t+1})
\end{bmatrix} = \left(H_{t+1}^{-1} \begin{bmatrix}
F_{1t+1}(A_{1,t+1}) \\
-F_{1t+1}(A_{2,t+1})
\end{bmatrix} \right)' \begin{bmatrix}
K_{1t}(A_{1,t+1}) \\
K_{2t}(A_{2,t+1})
\end{bmatrix} \div \begin{bmatrix}
K_{2t}(A_{2,t+1}) \\
K_{1t}(A_{1,t+1})
\end{bmatrix}
\]  \hspace{1cm} (18)

where \(./\) refers to element-by-element division.

When \(A_{1,t+1} + A_{2,t+1} > A_{t+2} - 1\), we eliminate \(A_{1,t+1} + A_{2,t+1} - A_{t+2} + 1\) columns of \(F\) to form a square matrix of rank \(A_{t+2} - 1\). This can be accomplished by successively eliminating linearly dependent columns, or by checking the rank of square matrices that result from removing arbitrary combinations of \(A_{1,t+1} + A_{2,t+1} - A_{t+2} + 1\) columns. Having obtained a square matrix satisfying the rank condition, we remove the corresponding elements of the matrix containing \(\Omega_{2,t+1}\) in (16) to make the matrices conformable; that is, we delete the elements that would have been multiplied by the columns removed from \(H_{t+1}\). This step sets the weight on the second action for the removed elements to zero. Following this procedure, an analogous equation to (18) is solved for the weights characterizing finite dependence.\(^{16}\)

### 3.3 Establishing \(\rho\)-period finite dependence when there are \(J\) choices

We now extend our framework to establish whether finite dependence exists after \(\rho\) periods rather than one and where the number of choices is now \(J\). More precisely, given specified decision weights between \(t + 1\) and \(t + \rho - 1\), two initial choices \(i\) and \(j\) in Equation (10) relabelled as 1 and \(J\).

\(^{16}\)The set of weights generated by this procedure depends on which linearly dependent columns are removed. Therefore the weight vectors satisfying finite dependence are not unique, but are linear transformations of each other.
2 for convenience, and an initial state $x_t$, we now provide a new set of necessary and sufficient conditions for whether $\kappa_{t+\rho}(x_{t+\rho+1}|x_t, 1) = \kappa_{t+\rho}(x_{t+\rho+1}|x_t, 2)$. For expositional simplicity we focus on optimization problems in which all the state variables are endogenous; later, however, our analysis of finite dependence for games shows that exogenous processes have no bearing on whether finite dependence exists or not, and when it does, play no role in determining its length or the weights.

Analogous to the one-period finite dependence case, for any $\tau \in \{t + 1, \ldots, t + \rho - 1\}$ we say $x \in \{1, \ldots, X\}$ is attainable by a sequence of decision weights from initial choice $k \in \{1, 2\}$ if the weight on $x$ is nonzero. Let $A_{j\tau} \in \{1, \ldots, X\}$ denote the number of attainable states, and $A_{j\tau} \subseteq \mathcal{X}$ the set of attainable states for the sequence beginning with choice $j$. Similarly let $A_{\tau+1} \in \{1, \ldots, X\}$ denote the number of states that are attainable by at least one of the sequences beginning either with choice 1 or 2, and denote by $A_{\tau+1} \subseteq \mathcal{X}$ the corresponding set.

Given an initial state and choice, we denote by $F_{k\tau}(A_{j\tau})$ the first $A_{\tau+1} - 1$ columns of the $A_{j\tau} \times A_{\tau+1}$ the transition matrix from $A_{j\tau}$ to $A_{\tau+1}$ when $k$ is chosen at period $\tau$. The matrix comprises elements $f_{k\tau}(x'|x)$ for each $x \in A_{j\tau}$ and $x' \in A_{\tau+1}$. Finally form the $(A_{\tau+1} - 1) \times (J - 1) [A_{1\tau} + A_{2\tau}]$ matrice $H_{\tau-t}$ from the $F_{k\tau}(A_{j\tau})$ matrices:

$$H_{\tau} = \begin{bmatrix}
F_{2\tau}(A_{2\tau}) - F_{1\tau}(A_{2\tau}) \\
\vdots \\
F_{J\tau}(A_{2\tau}) - F_{1\tau}(A_{2\tau}) \\
F_{1\tau}(A_{1\tau}) - F_{2\tau}(A_{1\tau}) \\
\vdots \\
F_{1\tau}(A_{1\tau}) - F_{J\tau}(A_{1\tau})
\end{bmatrix}$$

(19)

Our characterization of $\rho$-period dependence hinges on the rank of $H_{\tau}$.

**Theorem 2** Finite dependence from $x_t$ with respect to choices $i$ and $j$ can be achieved in $\rho = \tau - t$ periods for a given set of weights if and only if there exists decision weights from $t + 1$ to $\tau - 1$ such
that the rank of $H_\tau$ is $A_{t+\rho} - 1$.

There are an infinite number of weighting schemes, each of which might conceivably establish finite dependence. This fact explains why researchers have opted for guess and verify methods when designing models that exhibit this computationally convenient property. However, our next theorem, proved by construction in the Appendix, shows that an exhaustive search for a set of weights that establish finite dependence can be achieved in a finite number of steps. The key to the proof is that although the definition of $H_\tau$ does indeed depend on the weights, many sets of weights produce the same $A_{1\tau}$ and $A_{2\tau}$ (and hence the same $A_{\tau+1}$). Since the inversion of $H_\tau$ hinges on the attainable states, and the sets of all possible attainable states is finite, a finite number of operations is needed to establish whether a finite dependence path exists.

**Theorem 3** For each $\tau \in \{t+1, \ldots, \rho\}$ the rank of $H_\tau$ can be determined in a finite number of operations.

Having satisfied the rank condition, we can obtain the weights for the whole sequence by first eliminating linearly dependent columns from (19) when the number of columns in the linear system is greater than the number of rows to obtain a square matrix with the same rank, and then inverting.

While Theorem 3 applies to the full class of finite dependence problems, the number of calculations will be application-specific. As $\rho$ increases, so too will the sets of possible attainable states, increasing computational complexity in finding the finite dependence path. Increasing the number of choices, $J$, also will increase the sets of possible attainable states. At the same time, increasing $J$ gives more control to line up the states. When examining finite dependence for a pair of initial choices, the minimum $\rho$ must be weakly decreasing as more choices are available as one could always set the weight on these additional choices to zero. Finally, the complexity of the state space does not necessarily lead to more calculations to determine finite dependence for two reasons. First, it is only the states that can be reached in $\rho$ periods from the current state that are relevant for determining
finite dependence. Second, as the number of attainable states increases, the researcher also has more options for finding finite dependence paths due to being able to set the weights associated with each state.

### 3.4 Finite dependence in games

The methods developed above are directly applicable to dynamic games off short panels, that is, after modifying the notation with the \((n)\) superscripts as appropriate. Nevertheless establishing finite dependence in games is more onerous. Finite dependence in a game is player specific; in principle finite dependence might hold for some players but not for others. Furthermore, the transitions of the state variables depend on the decisions of all the players, not just player \(n\). Thus, finite dependence in games is ultimately a property that derives not just from the game primitives, but is defined with respect to an equilibrium. For this reason games of incomplete information generally do not exhibit one period finite dependence. If two alternative choices of \(n\) at time \(t\) affect the equilibrium choices the other players make in the next period at \(t + 1\) (or later), it is generally not feasible to line up the states across both paths emanating from the respective choices by the beginning of period \(t + 2\).

The existence of finite dependence in games for a given player \(n\) can be established if two conditions are met by the model and the equilibrium played out in the data. First, by taking a sequence of weighted actions player \(n\) can induce equilibrium play by the others so that after a finite number of periods, say \(\rho\), the distribution of \(x_{t+\rho+1}^{(\sim n)}\), conditional on \(x_{t}^{(\sim n)}\), does not depend on whether the sequence started with the choice \(j\) or \(k\). Whether this condition is satisfied or not depends on the reduced form transitions \(f_{t}^{(\sim n)}\left(x_{t+1}^{(\sim n)} \mid x_{t}^{(\sim n)}\right)\). Second, given the distribution of states for player \(n\) at \(t + \rho\) from the two sequences, one period finite dependence applies to \(x_{t+\rho+1}^{(n)}\), meaning that the player is able to line up his own state after executing the weighted sequences across the two paths to line up the states of the other players. This condition is determined by primitives alone, namely the \(f_{jt}^{(n)}\left(x_{t+1}^{(n)} \mid x_{t}^{(n)}\right)\) matrices.
3.5 Establishing finite dependence in games

To formally establish finite dependence in games settings, first note from (10) that the definition of finite dependence at $\tau$ for this class of games requires:

$$
\sum_{x_{\tau}=1}^{X} \sum_{k=1}^{J} f^{(n)}_{x_{\tau+1}}(x_{\tau}) f^{(n)}_{x_{\tau+1}}(x_{\tau}) \left[ \sum_{k=1}^{J} \omega^{(n)}_{k\tau}(x_{\tau},j) f^{(n)}_{x_{\tau+1}}(x_{\tau}|x_{\tau},j) - \omega^{(n)}_{k\tau}(x_{\tau},j') f^{(n)}_{x_{\tau+1}}(x_{\tau}|x_{t},j') \right] = 0
$$

(20)

We provide a set of sufficient conditions for (20) to hold that are relatively straightforward to check. They are based on the intuition that from periods $t+1$ through $\tau-1$ player $n$ takes actions that indirectly induce the other players to align $x^{(n)}_{\tau+1}$ through their equilibrium choices, and that at date $\tau$ player $n$ takes an action that aligns $x^{(n)}_{\tau+1}$. A necessary condition for $\tau$ dependence for $n$ as it relates to the other players can be derived by summing over the $x^{(n)}_{\tau+1}$ outcomes in (20).

Noting that:

$$
\sum_{x_{\tau}=1}^{X} \sum_{k=1}^{J} f^{(n)}_{x_{\tau+1}}(x_{\tau}) \left[ \sum_{k=1}^{J} \omega^{(n)}_{k\tau}(x_{\tau},j) f^{(n)}_{x_{\tau+1}}(x_{\tau}|x_{\tau},j) \right] \kappa^{(n)}_{\tau-1}(x_{\tau}|x_{t},j) = 0
$$

(21)

From the definition of (20), whether (21) holds or not only depends on the weights assigned to $n$ in periods $t+1$ though $\tau-1$, but not on the weights chosen in period $\tau$.

To derive a rank condition under which (21) holds, it is notationally convenient to focus on the first two choices as before. Suppose (21) holds at $\tau+1$. Then there must be decision weights at $\tau-1$ with the following property: the states that result in $\tau$ lead the other players to make (equilibrium)
decisions at $\tau$ so that each of their own states have the same weight across the two paths at $\tau + 1$.

Formally, denote by:

1. $A_{\tau}^{(n)} \subseteq X$ the set of attainable states at $\tau - 1$ given the choice sequence beginning with $j$ by player $n$.

2. $A_{\tau}^{(n)} \subseteq X$ the set of attainable states at $\tau$ given the choice sequence beginning with either 1 or 2 by player $n$.

3. $B_{\tau+1}^{(n)}$ the set of attainable states of the other players at $\tau + 1$ given the sequence beginning with either 1 or 2, $B_{\tau+1}^{(n)} \subseteq X^{(\sim n)}$

4. $B_{\tau+1}^{(n)}$ the number of elements in $B_{\tau+1}^{(n)}$.

5. $F_{k\tau-1}^{(n)}(A_{j\tau-1}^{(n)})$ the transition matrix from $A_{j\tau-1}^{(n)}$ to $A_{\tau}^{(n)}$ given choice $k$ at time $\tau - 1$ with competitors playing their equilibrium strategies.

6. $P_{\tau}^{(\sim n)}(A_{\tau}^{(n)})$ the transpose of the first $B_{\tau+1} - 1$ columns of the transition matrix from $A_{\tau}^{(n)}$ to the set of competitor states $B_{\tau+1}^{(n)}$.

Finally define $H_{\tau}^{(\sim n)}$ as:

$$H_{\tau}^{(\sim n)} \equiv P_{\tau}^{(\sim n)}(A_{\tau}^{(n)})$$

$$= \begin{bmatrix} F_{2\tau-1}^{(n)}(A_{2\tau-1}^{(n)}) - F_{1\tau-1}^{(n)}(A_{2\tau-1}^{(n)}) \\ \vdots \\ F_{J\tau-1}^{(n)}(A_{J\tau-1}^{(n)}) - F_{1\tau-1}^{(n)}(A_{J\tau-1}^{(n)}) \\ F_{1\tau-1}^{(n)}(A_{1\tau-1}^{(n)}) - F_{2\tau-1}^{(n)}(A_{1\tau-1}^{(n)}) \\ \vdots \\ F_{1\tau-1}^{(n)}(A_{1\tau-1}^{(n)}) - F_{J\tau-1}^{(n)}(A_{1\tau-1}^{(n)}) \end{bmatrix}$$

Games differ from the single agent setting because in a game the decisions at $\tau$ by the $n^{th}$ player’s rivals depend on the decision of $n$ at $\tau - 1$. Attaining finite dependence requires a form
of congruence between the effects the decision of \( n \) at \( \tau - 1 \) on the equilibrium decisions of other players at \( \tau \), and the subset of states controlled by player \( n \) attainable in one period.

First, finite dependence requires weighting rules at \( \tau - 1 \) so that when the other players take equilibrium actions at \( \tau \) on the two paths the states of the other players are lined up at \( \tau + 1 \). The effects of these equilibrium actions on the state operate through \( P_\tau^{(\sim n)} (A_\tau^{(n)}) \) in (22). Equation (22) then parallels equation (19) for the games setting where the states to be matched at \( \tau + 1 \) are the states of the other players rather than the state of the decision-maker. Following the same logic as that of Theorem 2 yields one set of necessary conditions for finite dependence in games.

Second, finite dependence requires that amongst the actions that player \( n \) might take to align \( x_{\tau+1}^{(n)} \), there is a weighted combination of actions by \( n \) produce a distribution of outcomes for \( x_{\tau+1}^{(n)} \) that are independent of \( x_\tau^{(n)} \). Formally there exist weights \( \omega_{k\tau}^{(n)}(x_\tau) \) and a weight distribution \( f_{\omega\tau}^{(n)}(x_{\tau+1}^{(n)}) \) such that:

\[
f_{\omega\tau}^{(n)}(x_{\tau+1}^{(n)}) = \sum_{k=1}^{J} \omega_{k\tau}^{(n)}(x_\tau) f_{k\tau}^{(n)}(x_{\tau+1}^{(n)} | x_\tau^{(n)})
\]  

(23)

Intuitively \( f_{\omega\tau}^{(n)}(x_{\tau+1}^{(n)}) \) is the distribution of time dependent mixture for one period finite dependence as applied to \( x_\tau^{(n)} \). The mixture over the choices of \( n \) at \( \tau \) can change with the state \( x_\tau \), but the distribution of \( x_{\tau+1}^{(n)} \) outcomes generated cannot.\(^{17}\)

Combining the two conditions (21) and (23) guarantees finite dependence because:

\[
0 = f_{\omega\tau}^{(n)}(x_{\tau+1}^{(n)}) \sum_{x_\tau=1}^{X} f_{\tau}^{(\sim n)}(x_{\tau+1}^{(\sim n)} | x_\tau^{(\sim n)}) \left[ \kappa_{\tau-1}^{(n)}(x_\tau | x_t, j) - \kappa_{\tau-1}^{(n)}(x_\tau | x_t, j') \right]
\]

\[
= \sum_{x_\tau=1}^{X} \sum_{k=1}^{J} f_{\tau}^{(\sim n)}(x_{\tau+1}^{(\sim n)} | x_\tau) f_{k\tau}^{(n)}(x_{\tau+1}^{(n)} | x_\tau^{(n)}) \left[ \omega_{k\tau}^{(n)}(x_\tau, j) \kappa_{\tau-1}^{(n)}(x_\tau | x_t, j) - \omega_{k\tau}^{(n)}(x_\tau, j') \kappa_{\tau-1}^{(n)}(x_\tau | x_t, j') \right]
\]

Thus the following theorem gives sufficient conditions for finite dependence to hold in games for a given player, say \( n \).

\(^{17}\)Trivially models with terminal or renewal actions satisfy (23), demonstrated by placing a weight of one on the terminal or renewal action and zero on all other choices.
Theorem 4 If, given initial choices 1 and 2, the rank of (22) is \( B_{\tau+1}^{(n)} - 1 \), and there exists weights at \( \tau \) such that (23) holds, then \( \rho = \tau - t \) period finite dependence is attained.

4 Applications

We now give two examples of how to apply to our finite dependence representation. The first is a job search model. Establishing finite dependence in a search model would seem difficult given that there is no guarantee one will receive another job offer in the future if an offer is turned down today and hence lining up, for example, future experience levels would seem difficult. We show that our representation applies directly to this case. The second is a coordination game where we apply the results of Theorem 4 to show that we can achieve two-period finite dependence in a strategic setting.

4.1 A search model

The following simple search model shows why negative weights are useful in establishing finite dependence, and uses the algorithm to exhibit an even less intuitive path to achieve finite dependence.

Each period \( t \in \{1, \ldots, T\} \) an individual may stay home by setting \( d_{1t} = 1 \), or apply for temporary employment setting \( d_{2t} = 1 \). Job applicants are successful with probability \( \lambda_t \), and the value of the position depends on the experience of the individual denoted by \( x \in \{1, \ldots, X\} \). If the individual works his experience increases by one unit, and remains at the current level otherwise. The preference primitives are given by the current utility from staying home, denoted by \( u_{1t} (x_t) \), and the utility from working, \( u_{2t} (x_t) \). Thus the dynamics of the model arise only from accumulating job experience, while nonstationarities arise from time subscripted offer arrival weights.
4.1.1 Constructing a finite dependence path

We demonstrate this model satisfies one-period finite dependence by constructing two paths that generate the same probability distribution of \( x_{t+2} \) conditional on \( x_t \). One path is defined by the pair \((\omega_{2t}, \omega_{2t+1}) = (0, \lambda_t/\lambda_{t+1})\), the individual stays home in period \( t \) and with decision weight \( \lambda_t/\lambda_{t+1} \) applies for temporary employment in period \( t+1 \). Note that \( \lambda_t/\lambda_{t+1} \) may be greater than one, implying \( \omega_{1t+1} \) is less than zero on this path. The other path is \((\omega_{2t}, \omega_{2t+1}) = (1, 0)\), an employment application in period \( t \) followed by staying home in period \( t+1 \). The distribution of \( x_{t+2} \) from following either path is the same: \( x_{t+2} = x_t \) with probability \( f_{2t}(x_t|x_t) = 1 - \lambda_t \), and \( x_{t+2} = x_t + 1 \) with probability \( f_{2t}(x_t + 1|x_t) = \lambda_t \).

Applying the finite dependence path, the difference in conditional value functions can then be expressed as:

\[
v_{2t}(x_t) - v_{1t}(x_t) = \lambda_t \left[ u_{2t}(x_t) - u_{1t}(x_t) + \beta u_{1t+1}(x_t + 1) - \beta u_{2t+1}(x_t) \right] + \beta \left[ \lambda_t \psi_1[p_{t+1}(x_t + 1)] + \lambda_t \left( \frac{1}{\lambda_{t+1}} - 1 \right) \psi_1[p_{t+1}(x_t)] - \frac{\lambda_{t+1}}{\lambda_t} \psi_2[p_{t+1}(x_t)] \right]
\]

4.1.2 Applying Theorem 2

While Section 4.1.1 provides a constructive example of forming a finite dependence path, it is also useful to show how the results from Section 3.2 apply. We now use the results from Section 3.2 to derive another finite dependence path.

To do so, we first define relevant terms in Equation (16). \( A_{1,t+1} \) and \( A_{2,t+1} \) are given by \( \{x_t\} \) and \( \{x_t, x_t + 1\} \) as if the individual chooses not to look for work the state remains unchanged while if the individual does not work he may either find employment or not. \( K_t(A_{1,t+1}) \) and \( K_t(A_{2,t+1}) \)
are then \([1]\) and \([1 - \lambda \lambda]'\). The relevant transition matrices are given by:

\[
F_{1,t+1}(A_{1,t+1}) = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{25}
\]

\[
F_{1,t+1}(A_{2,t+1}) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{26}
\]

\[
F_{2,t+1}(A_{1,t+1}) = \begin{bmatrix} 1 - \lambda_{t+1} & \lambda_{t+1} \end{bmatrix} \tag{27}
\]

\[
F_{2,t+1}(A_{2,t+1}) = \begin{bmatrix} 1 - \lambda_{t+1} & \lambda_{t+1} \\ 0 & 1 - \lambda_{t+1} \end{bmatrix} \tag{28}
\]

The last column, giving the transitions to state \(x_{t+1} = 2\), is omitted because if the probabilities are aligned in all but one attainable state, then the remaining probability must match up as well.

The system of equations in (16) has two equations—one for the probability of state \(x_t\) and the other for the probability of state \(x_{t+1}\)—and three choice variables. The three choice variables are the weights on the probability of choosing work conditional on either (i) work in the first period but no job \((x_{t+1} = x_t)\), (ii) work in the first period and obtaining a job \((x_{t+1} = x_t + 1)\), and (iii) not working in the first period \((x_{t+1} = x_t)\). We then have the following expression for the first term on the left-hand-side of (16):

\[
\left[ F_{2t+1}(A_{2,t+1}) - F_{1t+1}(A_{2,t+1}) \right]' = \begin{bmatrix} -\lambda_{t+1} & 0 & \lambda_{t+1} \\ \lambda_{t+1} & -\lambda_{t+1} & -\lambda_{t+1} \end{bmatrix} \tag{29}
\]

To reduce the system to two equations and two unknowns, we set the weight on looking for a job to zero conditional on being in state \(x_t\) at \(t + 1\) and having chosen not to look for work at \(t\). The last column of (29) can then be eliminated. The matrix we need to invert is then:

\[
\begin{bmatrix} -\lambda_{t+1} & 0 \\ \lambda_{t+1} & -\lambda_{t+1} \end{bmatrix}
\]
The solution to the system, given $\omega_{2,t+1}(x_t, 1) = 0$, is then:

$$
\begin{bmatrix}
\omega_{2,t+1}(x_t + 1, 2) \\
\omega_{2,t+1}(x_t, 1)
\end{bmatrix} =
\begin{bmatrix}
-1/\lambda_{t+1} & 0 \\
-1/\lambda_{t+1} & -1/\lambda_{t+1}
\end{bmatrix}
\begin{bmatrix}
\lambda_{t} \\
-\lambda_{t}
\end{bmatrix} =
\begin{bmatrix}
1 - \lambda_{t} \\
\lambda_{t}
\end{bmatrix} =
\begin{bmatrix}
\frac{-\lambda_{t}}{(1-\lambda_{t})\lambda_{t+1}}
\end{bmatrix}
$$

Finite dependence can then be achieved by setting $\omega_{2,t+1}(x_t, 1) = \omega_{2,t+1}(x_t+1, 2) = 0$ and $\omega_{2,t+1}(x_t, 2) = \frac{-\lambda_{t}}{(1-\lambda_{t})\lambda_{t+1}}$.

Note that here the path that begins with not looking for work involves not looking for work in period 2. By placing negative weight on looking for work conditional on (i) looking for work in period $t$ and (ii) not finding work at period $t$, we can cancel out the gains from successful search in period $t$. Hence we arrive at the state $x_t$ along both choice paths.

### 4.2 A coordination game

Because finite dependence in games requires lining up the distribution of one’s own states but also the states of one’s competitors, examples of finite dependence in the literature are scarce. One exception are models with exit decisions. Although finite dependence is usually not exploited in these models (but see Beauchamp, 2015 and Mazur, 2014), the models in Collard-Wexler (2013), Dunne et al. (2013), and Ryan (2012) all exhibit the finite dependence property that could be used to simplify estimation.

Our methods show finite dependence applies to a much broader class of games than those with terminal choices. To illustrate this, we provide an example of a two player coordination game. Each player $n \in \{1, 2\}$ chooses whether or not to compete in a market at time $t$ by setting $d_{2t}^{(n)} = 1$ if competing and setting $d_{1t}^{(n)} = 1$ if not. The dynamics of the game arise purely from the effect of decisions made by both players in the previous period on current payoffs; in this model $x_t = \{d_{2t-1}^{(1)}, d_{2t-1}^{(2)}\}$. Nonstationarity arises from the flow payoffs and corresponding choice probabilities rather than through the transitions on the state variables.

This model exhibits two period finite dependence. To prove this claim we find two sequences of
choices by the first player, which differ in their initial choice at \( t \), such that when the second player makes his equilibrium choices, the joint distribution of \( (d_{t+2}^{(1)}, d_{t+2}^{(2)}) \) is the same for both sequences. In this case, there is only one competitor state variable which is whether or not the competitor will be in the market at \( t + 2 \), so the rank condition is trivial to check. Further, we can ensure that player 1’s state is the same after the \( t + 2 \) decision by setting the \( t + 2 \) choice for player 1 to be the same across the two paths. Note that the choice of the first player at \( t + 2 \) has no effect on player 2’s choice at that time since it is not one of player 2’s state variables at \( t + 2 \). Theorem 5 establishes that a finite dependence path does indeed exist as well as specifying the finite dependence path.

**Theorem 5** Finite dependence for the two player coordination game can be achieved after two periods for all \( x_t \). Denote:

\[
R_1 = p_{2t+2}^{(2)}(2, 1) - p_{2t+2}^{(2)}(1, 1) + p_{2t+1}^{(2)}(2, 2) \left[ p_{2t+2}^{(2)}(2, 2) + p_{2t+2}^{(2)}(1, 1) - p_{2t+2}^{(2)}(2, 1) - p_{2t+2}^{(2)}(1, 2) \right]
\]

\[
R_2 = p_{2t+2}^{(2)}(2, 1) - p_{2t+2}^{(2)}(1, 1) + p_{2t+1}^{(2)}(2, 1) \left[ p_{2t+2}^{(2)}(2, 2) + p_{2t+2}^{(2)}(1, 1) - p_{2t+2}^{(2)}(2, 1) - p_{2t+2}^{(2)}(1, 2) \right]
\]

\( R_1 \) and \( R_2 \) cannot be zero. If \( R_1 \neq 0 \) then the path \( \omega_{2t+1}^{(1)}(1, 1) = \omega_{2t+1}^{(1)}(1, 2) = \omega_{2t+1}^{(1)}(2, 1) = 0 \) and

\[
\omega_{2t+1}^{(1)}(2, 1) = P_{t+2}^{(\sim n)} (A_{t+2}^{(n)}) \left[ \begin{array}{c}
F_{1t+1}^{(n)} (A_{1,t+1}^{(n)}) \\
-F_{1t+1}^{(n)} (A_{2,t+1}^{(n)})
\end{array} \right] \cdot \left( R_{2p2t}^{(2)} (x_t) \right)
\]

followed by setting the choice to zero at \( t + 2 \) in all states satisfies finite dependence. If \( R_1 = 0 \) then the path \( \omega_{2t+1}^{(1)}(1, 1) = \omega_{2t+1}^{(1)}(1, 2) = \omega_{2t+1}^{(1)}(2, 2) = 0 \) and

\[
\omega_{2t+1}^{(1)}(2, 1) = P_{t+2}^{(\sim n)} (A_{t+2}^{(n)}) \left[ \begin{array}{c}
F_{1t+1}^{(n)} (A_{1,t+1}^{(n)}) \\
-F_{1t+1}^{(n)} (A_{2,t+1}^{(n)})
\end{array} \right] \cdot \left( R_{2p2t}^{(2)} (x_t) \right)
\]

followed by setting the choice to zero at \( t + 2 \) in all states satisfies finite dependence.

5 Conclusion

CCP estimators provide a computationally cheap way to estimate dynamic discrete choice models in both single-agent and multi-agent settings. This paper precisely delineates and expands the class
of models that exhibit the finite dependence property used in CCP estimators, whereby only a-few-period-ahead conditional choice probabilities are used in estimation. Our approach applies a wide class of problems lacking stationarity, and is free of assumptions about the structure of the model and the beliefs of players regarding events that occur after the (short) panel has ended. For example these methods enable estimation of nonstationary infinite horizon games even when there are no terminal or renewal actions. Finally, when finite dependence does hold, there is no presumption that there is a unique set of weights defining finite dependence, a point illustrated in the search example. This raises the question about which set of weights should be used in estimation, a topic we defer to future research.\textsuperscript{18}

6 Appendix: Proofs

Proof of Theorem 1. With (bounded) negative weights the finite horizon results of Theorem 1 of Arcidiacono and Miller (2011) is easily adapted, since the proof of whether the positivity or negativity of the weights is not used in that proof.

Proof of Theorem 2. To complete the proof, we follow the approach laid out in the text for one period finite dependence case when there are only two choices. Define $K_{\tau-1}(A_{j\tau})$ as an $A_{j\tau}$ vector containing the probabilities of transitioning to each of the $A_{j\tau}$ attainable states given the choice sequence beginning with $j$ and state $x_t$. Denote $\Omega_{k\tau}(A_{j\tau})$ as a vector giving the weight placed on choice $k \in [1, \ldots, J]$ for each of the $A_{j\tau}$ possible states at $\tau$. Let $D_{j\tau}(A_{j\tau})$ be a $(J - 1)A_{j\tau}$ vector

\textsuperscript{18}Weighting future utility terms differently affects the asymptotic covariance matrix of the estimator, as well as its finite sample properties. Consequently choosing amongst alternative weighting schemes that attain finite dependence is application specific.
defined by:
\[
D_{j\tau}(A_{j\tau}) = \begin{bmatrix}
\Omega_{2\tau}(A_{j\tau}) \circ K_{\tau-1}(A_{j\tau}) \\
\vdots \\
\Omega_{k\tau}(A_{j\tau}) \circ K_{\tau-1}(A_{j\tau}) \\
\vdots \\
\Omega_{J\tau}(A_{j\tau}) \circ K_{\tau-1}(A_{j\tau})
\end{bmatrix}
\]
where \(\circ\) refers to element-by-element multiplication.

The \(A_{\tau+1}\) system of equations we need to solve can then be expressed as:
\[
H_{\tau} \begin{bmatrix}
D_{2\tau}(A_{2\tau}) \\
D_{1\tau}(A_{1\tau})
\end{bmatrix} = F_{1\tau}(A_{1\tau})K_{\tau-1}(A_{1\tau}) - F_{1\tau}(A_{2\tau})K_{\tau-1}(A_{2\tau}) \tag{30}
\]

Note that one of the equations is redundant because if all other states have the same weight assigned to them across the two paths then the last one must be lined up as well, implying that if the rank of \(H_{\tau}\) is \(A_{\tau+1} - 1\) then finite dependence holds in \(\rho\) periods. ■

**Proof of Theorem 3.** Define \(S\) by its components \(s = (s_1, \ldots, s_X)\) for all \(s_x \in \{0, 1\}\). Thus \(S\) represents the set of all the subsets of \(\{1, \ldots, X\}\). For convenience we now represent the state \(x\) in binary form as an element of \(S\), by setting \(s_x = 1\) and \(s_{x'} = 0\) for all \(x' \neq x\). Let \(s_{j,t+1}(x_t) \in S\) denote the attainable states in period \(t + 1\) when action \(j\) is taken at state \(x_t\) in period \(t\), in other words the strictly positive points of support of \(f_{j,t+1}(x_t)\). Similarly let \(S_{j,t+2}(x_t) \subseteq S\) comprise elements \(s_{j,t+2}(x_t) \in S_{j,t+2}(x_t)\) that arise with nonzero weight in period \(t + 2\) when action \(j\) is taken at state \(x_t\) in period \(t\) and then any weighted choice is made in period \(t+1\). We call \(s_{j,t+2}(x_t)\) attainable if every state in it arises with nonzero probability, and the actions taken to reach any state in the set cannot result in reaching a state outside the set. Proceeding inductively, for all finite \(\tau > t + 1\), let \(S_{j,\tau+1}(x_t) \subseteq S\) comprise elements \(s_{j,\tau+1}(x_t)\) that can be reached from some \(s_{j\tau}(x_t) \in S_{j\tau}(x_t)\).

We denote by \(R_{\tau+1}(s) \subseteq S\) the attainable subsets of \(S\) at period \(\tau+1\) in one period from \(s \in S\),
and first show that only a finite number of operations are required to define \( \mathcal{R}_{\tau+1} (x) \) for any \( x \in \mathcal{S} \). To check whether any \( r \in \mathcal{S} \) belongs to \( \mathcal{R}_{\tau+1} (x) \), without loss of generality we drop columns in the transition matrix that are linear combinations of the other columns to yield a matrix of \( X \times J (x) \) where \( J (x) \) is the number of linearly independent columns, and relabel the remaining choices by \( j \in \{ 1, \ldots, J (x) \} \). We now define for each \( \{ 1, \ldots, J (x) \} \) the real valued \( d_k \) satisfying the restriction \( \sum_{k=1}^{J(x)} d_k = 1 \). Then \( d \equiv (d_1, \ldots, d_{J(x)}) \) is a weighted choice mixture that induces, at the beginning of period \( \tau + 1 \), the weight distribution of states:

\[
\begin{bmatrix}
  f_{1\tau} (1 \mid x) & \cdots & f_{J(x),\tau} (1 \mid x) \\
  \vdots & \ddots & \vdots \\
  f_{1\tau} (X \mid x) & \cdots & f_{J(x),\tau} (X \mid x)
\end{bmatrix}
\begin{bmatrix}
  d_1 \\
  \vdots \\
  d_{J(x)}
\end{bmatrix} = F_\tau (x) d
\]  

(31)

Let \( r (x,d) \equiv (r_1 (x,d), \ldots, r_X (x,d)) \) indicate the states that have a nonzero weight on them. That is:

\[
\begin{align*}
  r_y (x, d) &\equiv 1 \left\{ \sum_{k=1}^{J(x)} d_k f_{k\tau} (y \mid x) \neq 0 \right\} \\
\end{align*}
\]

By definition \( r (x,d) \in \mathcal{R}_{\tau+1} (x) \).

We are now in a position to determine whether \( r \equiv (r_1, \ldots, r_X) \in \mathcal{S} \) belongs to \( \mathcal{R}_{\tau+1} (x) \) or not. Note that every \( r \in \mathcal{S} \) partitions the rows of \( F_\tau (x) \) into two, depending on whether row \( x' \) is assigned \( r_{x'} = 0 \) or \( r_{x'} = 1 \). Suppose \( r_{x'} = 0 \) but row \( x' \) of \( F_\tau (x) \) is a linear combination of rows indicated by \( s' \) where \( s_y = 1 \) for all \( y \in s' \). Then \( r \notin \mathcal{R}_{\tau+1} (x) \). Similarly if \( r_{x''} = 1 \) but row \( x'' \) is a linear combination of rows indicated by \( s'' \) where \( s_y = 0 \) for all \( y \in s'' \), then \( r \notin \mathcal{R}_{\tau+1} (x) \). Finally \( r \notin \mathcal{R}_{\tau+1} (x) \) if the number of linearly independent rows assigned a value of zero is greater than

\[\text{We call } (w (1 \mid x,d), \ldots, w (X \mid x,d)) \text{ a weight distribution because although } \sum_{x'=1}^{X} w (x' \mid x,d) = 1, \text{ there is no presumption that } w (x' \mid x,d) \geq 0, \text{ and hence it is not necessarily a probability distribution.}\]
Otherwise we can choose $d$ to solve:

$$
\begin{bmatrix}
  f_{1\tau}(1|x)(1-r_1) & \cdots & f_{J(x),\tau}(1|x)(1-r_1) \\
  \vdots & \ddots & \vdots \\
  f_{1\tau}(X|x)(1-r_X) & \cdots & f_{J(x),\tau}(X|x)(1-r_X)
\end{bmatrix}
\begin{bmatrix}
  d_1 \\
  \vdots \\
  d_{J(x)}
\end{bmatrix} = 0
$$

and in that way construct a weighted choice mixture to establish $r \in \mathcal{R}_{\tau+1}(x)$. Since there are $X!$ values of $r \in \mathcal{S}$, and only a finite number of determinants to evaluate when checking for linear independence, it follows that $\mathcal{R}_{\tau+1}(x)$ can be derived in a finite number of steps.\(^{20}\)

We recursively obtain $\mathcal{S}_{j,\tau+1}(x_t)$ from $\mathcal{S}_{j\tau}(x_t)$ using the $\mathcal{R}_{\tau+1}(x)$ sets. In the special case where $s \in \mathcal{S}_{j\tau}(x_t)$ is a singleton with $s = x$, it immediately follows that $\mathcal{R}_{\tau+1}(x) \subseteq \mathcal{S}_{j,\tau+1}(x_t)$. More generally $s'' \in \mathcal{S}_{j,\tau+1}(x_t)$ if and only if there exists an $s \in \mathcal{S}_{j\tau}(x_t)$, and an $s' \in \mathcal{R}_{\tau+1}(x)$ for each $x \in s$, such that $s'' \equiv \bigcup_{x \in s} s'$. By inspection only a finite number of operations are required to construct $\mathcal{S}_{j,\tau+1}(x_t)$ this way.

As in the two choice one period dependence case, only a finite number of operations are required to check the rank condition for each $(s, s') \in \mathcal{S}_{j,\tau+1}(x_t) \times \mathcal{S}_{j',\tau+1}(x_t)$, and there are only a finite number of combinations to check. ■

**Proof of Theorem 4.** The proof follows steps similar to that of Theorem 2. Define $K_{\tau-2}^{(n)}(A_{j\tau-1}^{(n)})$ as an $A_{j\tau-1}^{(n)}$ vector containing the probabilities of transitioning to each of the $A_{j\tau-1}^{(n)}$ attainable states given the choice sequence beginning with $j$ by player $n$ and state $x_t$. Denote $\Omega_{k\tau-1}^{(n)}(A_{j\tau-1}^{(n)})$ as a vector giving the weight placed on choice $k \in [1, \ldots, J]$ by player $n$ for each of the $A_{j\tau-1}^{(n)}$ possible

\(^{20}\)In practice, Gaussian elimination can be used to compute determinants. Here we are proving that only a finite number of steps are required, a step in establishing that an algorithm exists.
states at \( \tau - 1 \). Let \( D_{j\tau-1}^{(n)} \left( A_{j\tau-1}^{(n)} \right) \) be a \((J - 1)A_{j\tau-1}^{(n)}\) vector defined by:

\[
D_{j\tau-1}^{(n)} \left( A_{j\tau-1}^{(n)} \right) = \begin{bmatrix}
\Omega_{2\tau-1}^{(n)} \left( A_{2\tau-1}^{(n)} \right) \circ K_{\tau-2}^{(n)} \left( A_{\tau-2}^{(n)} \right) \\
\vdots \\
\Omega_{j\tau-1}^{(n)} \left( A_{j\tau-1}^{(n)} \right) \circ K_{\tau-2}^{(n)} \left( A_{\tau-2}^{(n)} \right) \\
\vdots \\
\Omega_{\tau\tau-1}^{(n)} \left( A_{\tau\tau-1}^{(n)} \right) \circ K_{\tau-2}^{(n)} \left( A_{\tau\tau-1}^{(n)} \right)
\end{bmatrix}
\]

where \( \circ \) refers to element-by-element multiplication.

The matrix representation of the finite dependence condition given in (21) for state \( x_{\tau+1}^{(n)} \) is then given by the \( B_{\tau+1}^{(n)} \) system of equations:

\[
H_{\tau}^{(\sim n)} \begin{bmatrix}
D_{2\tau-1}^{(n)} \left( A_{2\tau-1}^{(n)} \right) \\
D_{1\tau-1}^{(n)} \left( A_{1\tau-1}^{(n)} \right)
\end{bmatrix} = P_{\tau}^{(\sim n)} \left( A_{\tau}^{(n)} \right) \begin{bmatrix}
F_{1\tau-1}^{(n)} \left( A_{1\tau-1}^{(n)} \right) K_{\tau-2}^{(n)} \left( A_{1\tau-1}^{(n)} \right) - F_{1\tau-1}^{(n)} \left( A_{2\tau-1}^{(n)} \right) K_{\tau-2}^{(n)} \left( A_{2\tau-1}^{(n)} \right)
\end{bmatrix}
\]

(32)

Note that one of the equations is redundant because if all other competitor states have the same weight assigned to them across the two paths then the last one must be lined up as well, Hence if the rank of \( H_{\tau}^{(\sim n)} \) is \( B_{\tau+1}^{(n)} - 1 \) then finite dependence holds. ■

**Proof of Theorem 5.** We first specify Equation (16) for the games case with two choices. Note that in this game the number of own-states for player 2 is two: in or out in the previous period.

\[
P_{t+2}^{(\sim n)} \left( A_{t+2}^{(n)} \right) \begin{bmatrix}
F_{2t+1}^{(n)} \left( A_{2,t+1}^{(n)} \right) - F_{1t+1}^{(n)} \left( A_{2,t+1}^{(n)} \right) \\
F_{1t+1}^{(n)} \left( A_{1,t+1}^{(n)} \right) - F_{2t+1}^{(n)} \left( A_{1,t+1}^{(n)} \right)
\end{bmatrix} = \begin{bmatrix}
\Omega_{2,t+1}^{(n)} \left( A_{2,t+1}^{(n)} \right) \circ K_{2t}^{(n)} \left( A_{2,t+1}^{(n)} \right) \\
\Omega_{2,t+1}^{(n)} \left( A_{1,t+1}^{(n)} \right) \circ K_{1t}^{(n)} \left( A_{1,t+1}^{(n)} \right)
\end{bmatrix}
\]

(33)

We begin by defining the terms in the above expression, eliminating the last row of \( P_{t+2}^{(\sim n)} \left( A_{t+2}^{(n)} \right) \) as if we match the weight placed on one state we will automatically match the weight placed on the other state.
\[ P_{t+2}^{(2)}(A_{t+2}) = \begin{bmatrix} p_{2t+2}^{(2)}(2, 2) & p_{2t+2}^{(2)}(2, 1) & p_{2t+2}^{(2)}(1, 2) & p_{2t+2}^{(2)}(1, 1) \end{bmatrix} \]

(34)

\[ \begin{align*}
F_{1t+1}^{(n)}(A_{1, t+1}) & = \left[ \begin{array}{cccc}
0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
p_{2t+1}^{(2)}(1, 2) & p_{2t+1}^{(2)}(1, 1) & -p_{2t+1}^{(2)}(2, 2) & -p_{2t+1}^{(2)}(1, 2) \\
p_{1t+1}^{(2)}(1, 2) & p_{1t+1}^{(2)}(1, 1) & -p_{1t+1}^{(2)}(2, 2) & -p_{1t+1}^{(2)}(1, 1) \\
\end{array} \right]
\end{align*} \]

(35)

\[ \begin{align*}
F_{2t+1}^{(1)}(A_{2, t+1}) - F_{1t+1}^{(1)}(A_{2, t+1}) & = \left[ \begin{array}{cccc}
p_{2t+1}^{(2)}(2, 2) & p_{2t+1}^{(2)}(2, 1) & -p_{2t+1}^{(2)}(1, 2) & -p_{2t+1}^{(2)}(1, 1) \\
p_{1t+1}^{(2)}(2, 2) & p_{1t+1}^{(2)}(2, 1) & -p_{1t+1}^{(2)}(1, 2) & -p_{1t+1}^{(2)}(1, 1) \\
-p_{2t+1}^{(2)}(2, 2) & -p_{2t+1}^{(2)}(2, 1) & p_{2t+1}^{(2)}(1, 2) & p_{2t+1}^{(2)}(1, 1) \\
-p_{1t+1}^{(2)}(2, 2) & -p_{1t+1}^{(2)}(2, 1) & p_{1t+1}^{(2)}(1, 2) & p_{1t+1}^{(2)}(1, 1) \\
\end{array} \right] \\
\end{align*} \]

(36)

\[ K_{2t}(A_{2, t+1}^{(n)}) = K_{1t}(A_{2, t+1}^{(n)}) = \begin{bmatrix} p_{2t}^{(2)}(x) \\
p_{1t}^{(2)}(x) \end{bmatrix} \]

(37)

We have choices over four decision weights: one for each of the possible states \{(2, 2), (2, 1), (1, 2), (1, 1)\} but only need to match one probability. Hence we set the probabilities of entering in the last three states to zero. As we will show, for some values of the conditional choice probabilities the rank condition will not be satisfied. But in this case, we can set the probabilities of entering in all but state \((2,1)\) to zero and the rank condition will be satisfied. That is, the rank condition must be satisfied for one of these cases (and possibly both).

When all the probabilities of entering are set to zero for all states but \((2,2)\), the rank condition needed is that \(P_{t+2}^{(\sim n)}(A_{t+2}^{(n)})\) times the first column of (36) does not equal zero. We denote the result of this multiplication as \(R_1\) where:

\[ R_1 = p_{2t+2}^{(2)}(2, 1) - p_{2t+2}^{(2)}(1, 1) + p_{2t+1}^{(2)}(2, 2) \left[ p_{2t+2}^{(2)}(2, 2) + p_{2t+2}^{(2)}(1, 1) - p_{2t+2}^{(2)}(2, 1) - p_{2t+2}^{(2)}(1, 2) \right] \]
Similarly defining $R_2$ as the results of multiplying $P_{t+2}^{(n)}(A_{t+2}^{(n)})$ by the second column of (36) yields:

$$R_2 = p_{2t+2}^{(2)}(2, 1) - p_{2t+2}^{(2)}(1, 1) + p_{2t+2}^{(2)}(2, 2) + p_{2t+2}^{(2)}(1, 1) - p_{2t+2}^{(2)}(2, 1) - p_{2t+2}^{(2)}(1, 2)$$

Note that the two expressions are the same except for the term multiplying the expression in parentheses. Since the player’s own state is assumed to affect the conditional choice probabilities, $p_{2t+1}^{(2)}(2, 1) \neq p_{2t+1}^{(2)}(2, 2)$, both expressions cannot be zero.

If $R_1 \neq 0$, we can set

$$\omega_{2t+1}^{(1)}(2, 2) = p_{t+2}^{(n)}(A_{t+2}^{(n)}) \left[ \begin{array}{c} F_{1t+1}^{(n)}(A_{1t+1}^{(n)}) \\ -F_{1t+1}^{(n)}(A_{2t+1}^{(n)}) \end{array} \right] \cdot \left[ R_{1} p_{2t}^{(2)}(x_t) \right] \quad (38)$$

and set $\omega_{2t+1}^{(1)}(1, 1) = \omega_{2t+1}^{(1)}(1, 2) = \omega_{2t+1}^{(1)}(2, 1) = 0$. Then, setting the choice to zero in all states at $t + 2$ gives the finite dependence path.

If $R_1 = 0$, we can set

$$\omega_{2t+1}^{(1)}(2, 1) = p_{t+2}^{(n)}(A_{t+2}^{(n)}) \left[ \begin{array}{c} F_{1t+1}^{(n)}(A_{1t+1}^{(n)}) \\ -F_{1t+1}^{(n)}(A_{2t+1}^{(n)}) \end{array} \right] \cdot \left[ R_{2} p_{2t}^{(2)}(x_t) \right] \quad (39)$$

and set $\omega_{2t+1}^{(1)}(1, 1) = \omega_{2t+1}^{(1)}(1, 2) = \omega_{2t+1}^{(1)}(2, 2) = 0$. Again, setting the choice to zero in all states at $t + 2$ gives the finite dependence path. ■

References


