Appendix

The Approximate Solution of Finite-Horizon Discrete Choice Dynamic Programming Models

Revisiting Keane and Wolpin (1994)

Philipp Eisenhauer

University of Bonn

May 21, 2018

Contents

A. Computational Details

The `respy` package (respy, 2018) provides the computational support for the project. Its online documentation is available at `http://respy.readthedocs.io` and thus I only outline the implementation details that are specific to the reported results and not discussed in the manuscript itself.

Optimization I use the NEWUOA algorithm (Powell, 2006). All tuning parameters are set to their default values. I use a diagonal scale-based preconditioner based on a gradient approximation. I set its minimum value to 0.00001.

Integration The solution and estimation of the model produces two types of integrals. I need to determine $E\max$ during the solution step and simulate the choice probabilities to construct the sample log-likelihood. I evaluate both using Monte Carlo integration. I use 200 random draws for the choice probabilities.

Differentiation The derivatives required for the preconditioning are approximated by forward finite differences with a step size of 0.0001.

Function Approximation The details for the $E\max$ interpolation are already discussed in the text. However, there are some additional complications.

- Agents are only allowed to obtain 10 additional years of education. Thus, there exist a number of inadmissible states in late periods. However, $\bar{V}_3$ is still included in the interpolation regression and assigned an ad hoc penalty of -40,000 as in Keane and Wolpin (1994). Results are not sensitive to the exact value as only about 5% of the states in later periods are affected.

- As noted in their correspondence with the editor, Keane and Wolpin (1994) drop the linear term of V_3 from the interpolation regression for the first parameterization due to reported collinearity problems. These are due to the small variation in the consumption value of schooling across states. I encounter the same problem and thus follow their lead.

I am indebted to several other open source tools among them `matplotlib` (Hunter, 2007) and `Vagrant` (Hashimoto, 2013).

A.1. Compute Machine

This study utilized the high-performance computational capabilities of the Acropolis Linux cluster hosted by Social Sciences Computing Services at the University of Chicago. `Acropolis.uchicago.edu` is a clustered Linux system maintained by Social Science Computing Services at the University of Chicago. The head node contains 80 CPUs, 2 TB of memory and 110 TB of storage. The compute nodes provide an additional 1,792 CPUs, 8 TB of memory, and 4 NVIDIA Tesla GPUs.

B. Additional Results

This section presents all my results for each of the parameterizations in Table B.1. The *exact solution* is constructed using 100,000 random draws for the evaluation of $E\max$ at all states. The *exact sample* refers to a set of 1,000 simulated agents based on the *exact solution*. As on overall measure of the approximation error, I use the root-mean-square error (RMSE) by comparing the choice probabilities in the *exact sample* to a newly simulated set of 1,000 agents based on the relevant alternative parameterization of the model.

B.1. Parameterizations

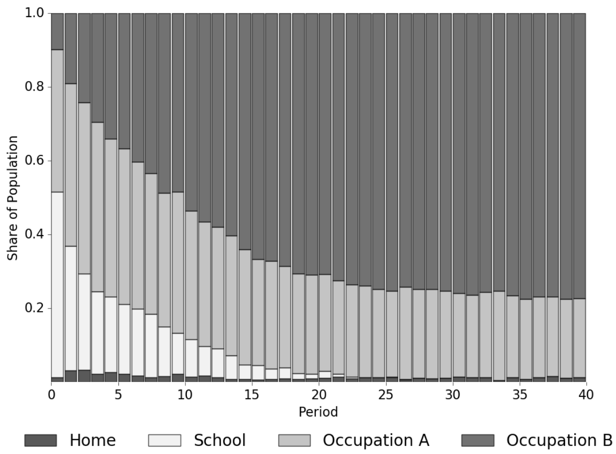
Table B.1: Parameterizations

Parameter	Data One	Data Two	Data Three
α_{10}	9.2100	9.2100	8.0000
α_{11}	0.0380	0.0400	0.0700
α_{12}	0.0330	0.0330	0.0550
α_{13}	0.0005	0.0005	0.0000
α_{14}	0.0000	0.0000	0.0000
α_{15}	0.0000	0.0000	0.0000
α_{20}	8.4800	8.2000	7.9000
α_{21}	0.0700	0.0800	0.0700
α_{22}	0.0670	0.0670	0.0600
α_{23}	0.0010	0.0010	0.0000
α_{24}	0.0220	0.0220	0.0550
α_{25}	0.0005	0.0005	0.0000
β_0	0.0000	5,000.0000	5,000.0000
β_1	0.0000	5,000.0000	5,000.0000
β_2	4,000.0000	15,000.0000	20,000.0000
γ_0	17,750.0000	14,500.0000	21,500.0000
$(\sigma_{11})^{1/2}$	0.2000	0.4000	1.0000
σ_{12}	0.0000	0.0000	0.5000
σ_{13}	0.0000	0.0000	0.0000
σ_{14}	0.0000	0.0000	0.0000
$(\sigma_{22})^{1/2}$	0.2500	0.5000	1.0000
σ_{23}	0.0000	0.0000	0.0000
σ_{24}	0.0000	0.0000	0.0000
$(\sigma_{33})^{1/2}$	1,500.0000	6,000.0000	7,000.0000
σ_{34}	0.0000	0.0000	-2.975×10^7
$(\sigma_{44})^{1/2}$	1,500.0000	6,000.0000	8,500.0000

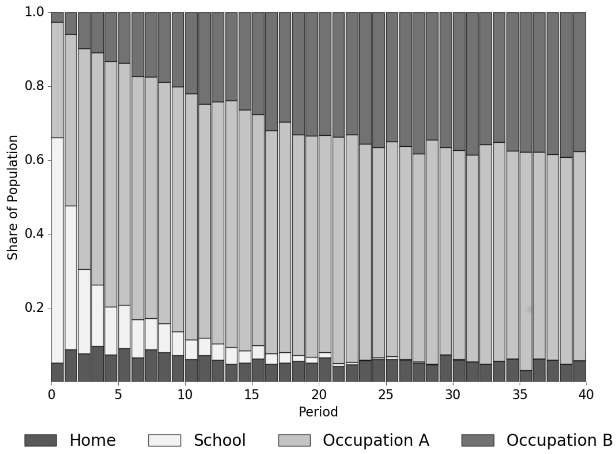
B.2. Choice Patterns

Figure B.1 shows the share of agents in the *exact sample* opting for each of the four alternatives by period.

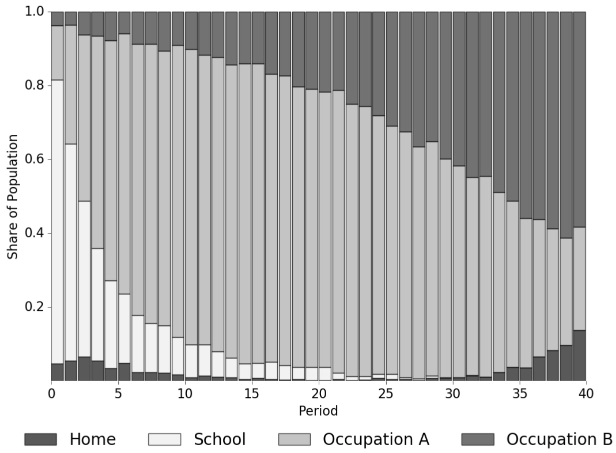
Figure B.1: Choice Patterns



(a) Data One



(b) Data Two



(c) Data Three

B.3. Correct Choices

Tables B.2 - B.4 show the proportion of correct choices for alternative interpolation schemes.

B.4. Monte Carlo Exercise

Tables B.5 - B.7 show the estimation performance for each of the model parameters during the initial Monte Carlo exercise. Let θ_i denote the true value of parameter i, $\hat{\theta}_i$ its average estimate across all Monte Carlo iterations, and $\hat{\theta}_{ij}$ the estimated parameter in iteration j. The statistics in the Table B.5 - B.7 are calculated as follows:

Bias	$\hat{\theta}_i - \theta_i$
t - statistic	$\left(\frac{\hat{\theta}_i - \theta_i}{\sigma_{\hat{\theta}_i}}\right)\sqrt{40}$
Standard Deviation	$\left[\frac{1}{39}\sum_{j=1}^{40}(\hat{\theta}_{ij} - \hat{\theta}_i)^2\right]^{\frac{1}{2}}$

Note that the table contains the Cholesky decomposition parameters a_{ij} of the covariance matrix of the shocks to the immediate rewards. I report the RMSE, the total number of evaluations of the criterion function, and the number of steps of the optimizer as their average across all 40 Monte Carlo iterations.

I specify 200 interpolation points, use 500 random draws for the evaluation of $E\max$, and allow for a maximum of 10,000 evaluations of the criterion function by the optimizer for each estimation.

Table B.2: Correct Choices, Dataset One

Points	All	All	All	2,000	500
E max Draws	2,000	1,000	250	2,000	2,000
			At Selected Periods		
Period					
1	1.000	0.998	0.938	0.967	0.942
10	0.990	1.000	0.989	0.988	0.979
20	1.000	1.000	1.000	0.998	0.999
30	1.000	1.000	1.000	0.994	0.998
40	1.000	1.000	1.000	1.000	1.000
Total	0.999	0.998	0.994	0.993	0.991
			Number of Periods over the Lifetime		
Periods					
1 - 10	0.000	0.000	0.000	0.000	0.000
11 - 35	0.000	0.000	0.000	0.000	0.000
36 - 38	0.001	0.000	0.020	0.027	0.046
39	0.036	0.043	0.181	0.190	0.252
40	0.963	0.957	0.799	0.783	0.702
Average	39.962	39.957	39.777	39.752	39.649

Table B.3: Correct Choices, Dataset Two

Points	All	All	All	2,000	500
$E\max$ Draws	2,000	1,000	250	2,000	2,000
At Selected Periods					
Period					
1	0.998	0.994	0.993	0.996	0.988
10	1.000	0.998	0.995	0.990	0.972
20	1.000	0.997	0.994	0.979	0.961
30	0.998	1.000	0.998	0.988	0.989
40	1.000	1.000	1.000	1.000	1.000
Total	0.998	0.997	0.995	0.990	0.981
Number of Periods over the Lifetime					
Periods					
1 - 10	0.000	0.000	0.000	0.000	0.000
11 - 35	0.000	0.000	0.000	0.001	0.001
36 - 38	0.003	0.003	0.012	0.062	0.157
39	0.040	0.085	0.172	0.260	0.361
40	0.957	0.912	0.816	0.677	0.481
Average	39.954	39.909	39.804	39.600	39.265

Table B.4: Correct Choices, Dataset Three

Points	All	All	All	2,000	500
$E\max$ Draws	2,000	1,000	250	2,000	2,000
At Selected Periods					
Period					
1	0.995	0.993	0.985	0.991	0.979
10	0.995	0.995	0.982	0.975	0.931
20	0.995	0.997	0.994	0.979	0.940
30	0.994	0.999	0.989	0.974	0.972
40	1.000	1.000	1.000	1.000	1.000
Total	0.995	0.995	0.991	0.980	0.959
Number of Periods over the Lifetime					
Periods					
1 - 10	0.000	0.000	0.000	0.000	0.000
11 - 35	0.000	0.000	0.000	0.003	0.030
36 - 38	0.015	0.015	0.038	0.187	0.432
39	0.142	0.150	0.249	0.324	0.304
40	0.843	0.835	0.713	0.486	0.234
Average	39.827	39.817	39.671	39.226	38.374

Table B.5: Monte Carlo Exercise, Dataset One

Parameter	True Value	Bias	t - statistic	Std. Deviation
α_{10}	9.2100	0.0011	1.5154	0.0047
α_{11}	0.0380	-0.0001	-0.9115	0.0005
α_{12}	0.0330	-0.0001	-1.6259	0.0004
α_{13}	0.0005	0.0000	-9.5394	0.0000
α_{14}	0.0000	-0.0019	-6.3082	0.0019
α_{15}	0.0000	0.0000	-2.5532	0.0001
α_{20}	8.4800	-0.0011	-3.6070	0.0019
α_{21}	0.0700	0.0000	-1.7435	0.0001
α_{22}	0.0670	0.0000	1.2959	0.0001
α_{23}	0.0010	0.0000	-2.0817	0.0000
α_{24}	0.0220	-0.0003	-3.6062	0.0005
α_{25}	0.0005	0.0000	-2.0721	0.0000
β_0	0.0000	-75.5007	-3.3125	144.1518
β_1	0.0000	25.5030	0.9254	174.2990
β_2	4,000.0000	27.8670	0.7521	234.3484
γ_0	17,750.0000	-24.0426	-0.7838	194.0135
a_{11}	0.2000	0.0003	0.3921	0.0045
a_{21}	0.0000	-0.0168	-4.3305	0.0245
a_{22}	0.2500	0.0009	1.4127	0.0040
a_{31}	0.0000	8.7165	0.2842	193.9909
a_{32}	0.0000	-74.6767	-2.5699	183.7807
a_{33}	1,500.0000	-209.4873	-5.9595	222.3199
a_{41}	0.0000	-30.4417	-1.2886	149.4149
a_{42}	0.0000	-161.8797	-3.1690	323.0728
a_{43}	0.0000	-51.5953	-0.9814	332.4948
a_{44}	1,500.0000	-175.2579	-3.8345	289.0676
Steps	546		Evaluations	2,015
RMSE	0.0577			

Notes: Std. Deviation = Standard Deviation. I calculate the number of steps, number of evaluations, and the RMSE as the average across all 40 Monte Carlo iterations.

Table B.6: Monte Carlo Exercise, Dataset Two

Parameter	True Value	Bias	t - statistic	Std. Deviation
α_{10}	9.2100	-0.0020	-3.0922	0.0042
α_{11}	0.0400	0.0000	-1.0542	0.0002
α_{12}	0.0330	-0.0001	-1.7489	0.0002
α_{13}	0.0005	0.0000	0.5340	0.0000
α_{14}	0.0000	-0.0001	-2.2285	0.0003
α_{15}	0.0000	0.0000	0.6925	0.0000
α_{20}	8.2000	-0.0068	-4.4297	0.0097
α_{21}	0.0800	-0.0002	-2.7728	0.0005
α_{22}	0.0670	-0.0002	-2.0449	0.0005
α_{23}	0.0010	0.0000	-1.1855	0.0000
α_{24}	0.0220	0.0002	2.0316	0.0006
α_{25}	0.0005	0.0000	-6.3460	0.0000
β_0	5,000.0000	-44.4702	-1.6556	169.8772
β_1	5,000.0000	110.2680	1.2149	574.0130
β_2	15,000.0000	-179.5200	-2.2953	494.6457
γ_0	14,500.0000	-96.5064	-2.7688	220.4454
a_{11}	0.4000	-0.0413	-1.4829	0.1762
a_{21}	0.0000	0.0064	1.8592	0.0217
a_{22}	0.5000	-0.0011	-1.0425	0.0068
a_{31}	0.0000	-102.9634	-3.0118	216.2179
a_{32}	0.0000	21.4127	0.3744	361.7239
a_{33}	6,000.0000	107.5406	2.5494	266.7858
a_{41}	0.0000	-14.0007	-0.2566	345.1081
a_{42}	0.0000	-97.9643	-2.0837	297.3506
a_{43}	0.0000	-103.7960	-1.6446	399.1552
a_{44}	6,000.0000	34.1112	0.9186	234.8650
Steps	19		Evaluations	430
RMSE	0.0303			

Notes: Std. Deviation = Standard Deviation. I calculate the number of steps, number of evaluations, and the RMSE as the average across all 40 Monte Carlo iterations.

Table B.7: Monte Carlo Exercise, Dataset Three

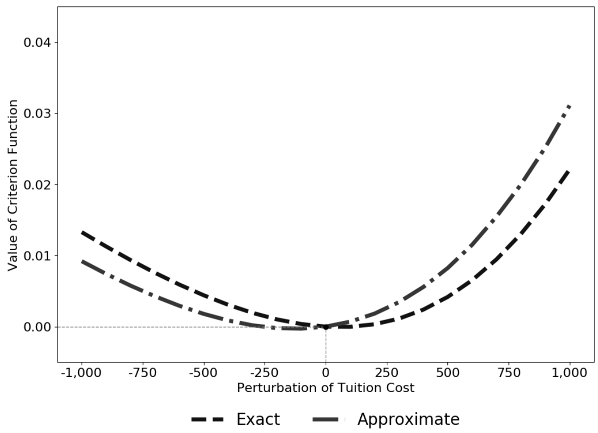
Parameter	True Value	Bias	t - statistic	Std. Deviation
α_{10}	8.0000	-0.0092	-5.3135	0.0110
α_{11}	0.0700	-0.0008	-3.3256	0.0016
α_{12}	0.0550	0.0006	1.8263	0.0020
α_{13}	0.0000	0.0000	0.0452	0.0001
α_{14}	0.0000	-0.0029	-5.111	0.0036
α_{15}	0.0000	-0.0006	-5.4264	0.0007
α_{20}	7.9000	-0.0039	-4.3200	0.0057
α_{21}	0.0700	0.0006	1.2129	0.0031
α_{22}	0.0600	-0.0017	-4.5783	0.0023
α_{23}	0.0000	0.0001	4.1808	0.0001
α_{24}	0.0550	0.0003	1.0699	0.0019
α_{25}	0.0000	0.0000	-2.4916	0.0001
β_0	5,000.0000	-40.5552	-0.3268	784.8442
β_1	5,000.0000	25.4629	0.2931	549.4277
β_2	20,000.0000	-9.4701	-0.0403	1,487.7457
γ_0	21,500.0000	19.3394	4.2987	28.4534
a_{11}	1.0000	-0.0230	-3.7447	0.0389
a_{21}	0.5000	0.0040	1.2691	0.0198
a_{22}	0.8660	-0.0303	-2.9467	0.0651
a_{31}	0.0000	-164.1444	-1.9570	530.4788
a_{32}	0.0000	722.0474	2.3833	1,916.1160
a_{33}	7,000.0000	786.9953	4.2248	1,178.1343
a_{41}	0.0000	-88.1003	-0.5460	1,020.4223
a_{42}	0.0000	-687.4966	-2.5796	1,685.5875
a_{43}	-4,250.0000	-672.5874	-2.2242	1,912.5056
a_{44}	7,361.2159	-112.1582	-0.4167	1,702.4584
Steps	17		Evaluations	421
RMSE	0.0249			

Notes: Std. Deviation = Standard Deviation. I calculate the number of steps, number of evaluations, and the RMSE as the average across all 40 Monte Carlo iterations.

B.5. Noise in Criterion Function

Figure B.2 shows the exact and approximate criterion function around the true parameter values. To get a sense of a possible discrepancy between the two, I perturb β_1 around its true value in \$100 increments. This parameter captures the effect of tuition cost on educational enrollment and is of particular interest for the ex ante evaluation of tuition policies.[1] While the exact criterion function has its minimum at the actual value, this is not true for the approximated function. The latter attains its minimum at a perturbation of -\$100. This casts doubt on the quality of the approximation.

Figure B.2: Criterion Functions



B.6. Interpolation Schemes

Table B.8 repeats the estimation results based on alternative interpolation schemes. I use a single processor to ensure comparability of computation times. Table B.9 shows the estimated parameter values from the misspecified static estimation that provides the starting values for the Monte Carlo Exercise. I solve the DP problem at all states and use 500 random draws for the evaluation of $E\max$. I allow for a maximum of 1,000 evaluations of the criterion function by the optimizer which terminates after 284 steps. Each Monte Carlo iteration starts with a RMSE of 0.16.

[1] See Keane and Wolpin (1997) and Keane and Wolpin (2001) for examples.

Table B.8: Interpolation Schemes

Points	200	500	1,500	All
E max Draws	500	500	500	500
RMSE	0.10	0.09	0.07	0.03
Minutes	35	96	361	794
Steps	1,090	2,671	5,730	32,774
Evaluations	3,017	6,412	12,020	59,680

Notes: All results are calculated as the average across all 40 Monte Carlo iterations.

C. Recomputation Instructions

I provide an image of a virtual machine (VM) for download to ensure full recomputability of my results. The image contains a software-based emulation of a computer, where all the required software is already pre-installed.

Two additional software tools are required: (1) `VirtualBox` and (2) `Vagrant`. `VirtualBox` is a virtualization software, while `Vagrant` provides a convenient wrapper around it. Both are free and open-source. Please consult their websites for installation instructions. The following instructions were tested for `VirtualBox 5.2` and `Vagrant 2.0`.

Once `VirtualBox` and `Vagrant` are available, the image can be downloaded and accessed by the following commands:

```
$ vagrant init structRecomputation/base
$ vagrant up --provider virtualbox
$ vagrant ssh
```

As all the required software is already installed, recomputation is straightforward. Simply typing the following command into the terminal produces all the results in the manuscript:

```
$ ./recompute
```

The output files will be available in the `_recomputed` subdirectory. The process takes several days due to the large number of Monte Carlo iterations for some of the results. I aligned the virtual machine to the project's compute server as much

Table B.9: Static Estimation

Parameter	True	Estimated
α_{10}	9.2100	9.2010
α_{11}	0.0380	0.0378
α_{12}	0.0330	0.0325
α_{13}	0.0005	0.0005
α_{14}	0.0000	-0.0048
α_{15}	0.0000	-0.0002
α_{20}	8.4800	8.8519
α_{21}	0.0700	0.0639
α_{22}	0.0670	0.0527
α_{23}	0.0010	0.0010
α_{24}	0.0220	0.0202
α_{25}	0.0005	0.0009
β_0	0.0000	-130.4283
β_1	0.0000	50.1235
β_2	4,000.0000	7,126.9279
γ_0	17,750.0000	17,667.9808
a_{11}	0.2000	0.2099
a_{21}	0.0000	0.0307
a_{22}	0.2500	0.2344
a_{31}	0.0000	-12,546.1078
a_{32}	0.0000	2,906.4830
a_{33}	1,500.0000	1,630.7701
a_{41}	0.0000	293.5752
a_{42}	0.0000	2,316.2633
a_{43}	0.0000	-95.3110
a_{44}	1,500.0000	1,592.5645
Steps	268	
Evaluations	1,000	

Table C.1: Mapping between Files and Results

File	Keane and Wolpin (1994)	Eisenhauer (2018)
	Correct Choices	
`table_2.1.txt`	Table 2.1	Table B.2
`table_2.2.txt`	Table 2.2	Table B.3
`table_2.3.txt`	Table 2.3	Table B.4
	Monte Carlo Estimation	
`table_4.1.txt`	Table 4.1	Table B.5
`table_4.2.txt`	Table 4.2	Table B.6
`table_4.3.txt`	Table 4.3	Table B.7
	Choice Patterns	
`choices_one.png`	—	Figure B.1
`choices_two.png`	—	Figure B.1
`choices_three.png`	—	Figure B.1
	Additional Results	
`schemes.txt`	—	Table B.8
`static.txt`	—	Table B.9
`graphs_criterions.png`	—	Figure B.2
`performance.txt`	—	Section 3.3

as possible, however small numerical discrepancies between the recomputed and published results are to be expected. The original results from the compute server are available in the `_published` subdirectory. There is a slight difference in the order and sign of the coefficients between the output files and the results in this paper, please see `respy`'s online documentation for details. Table C.1 provides the mapping between the output files and the results reported in the two relevant publications.

References

Eisenhauer, P. (2018). The approximate solution of finite-horizon dynamic programming models: Revisiting Keane and Wolpin (1994). *resubmitted to Journal of Applied Econometrics*.

Hashimoto, M. (2013). *Vagrant: Up and Running*. O'Reilly Media, Sebastopol, CA, 1st edition.

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering*, 9(3):90–95.

Keane, M. P. and Wolpin, K. I. (1994). The solution and estimation of discrete choice dynamic programming models by simulation and interpolation: Monte Carlo evidence. *Review of Economics and Statistics*, 76(4):648–672.

Keane, M. P. and Wolpin, K. I. (1997). The career decisions of young men. *Journal of Political Economy*, 105(3):473–522.

Keane, M. P. and Wolpin, K. I. (2001). The effect of parental transfers and borrowing constraints on educational attainment. *International Economic Review*, 42(4):1051–1103.

Powell, M. J. D. (2006). The NEWUOA software for unconstrained optimization without derivatives. In Pillo Gianni, R. M., editor, *Large-Scale Nonlinear Optimization*, Nonconvex Optimization and Its Applications, pages 255–297. Springer, Boston, MA.

respy (2018). respy: A Python package for the simulation and estimation of a prototypical discrete choice dynamic programming model.