

# Fast Bellman Iteration: An Application of Legendre-Fenchel Duality to Infinite-Horizon Dynamic Programming in Discrete Time\*

Ronaldo Carpio<sup>†</sup>      Takashi Kamihigashi<sup>‡</sup>

March 10, 2015

## Abstract

We propose an algorithm, which we call “Fast Bellman Iteration” (FBI), to compute the value function of an infinite-horizon dynamic programming problem in discrete time. FBI is an extremely efficient linear-time algorithm applicable to a class of multidimensional dynamic programming problems with concave return (or convex cost) functions and linear constraints. In this algorithm, a sequence of functions is generated starting from the zero function by repeatedly applying a simple algebraic rule involving the Legendre-Fenchel transform of the return function. The resulting sequence is guaranteed to converge, and the Legendre-Fenchel transform of the limiting function coincides with the value function.

---

\*Part of this research was conducted while Ronaldo Carpio was visiting RIEB, Kobe University, as a Visiting Researcher in 2015. Earlier versions of this paper were presented at the SCE Conference on Computing in Economics and Finance in Oslo, 2014, and the SAET Conference in Tokyo, 2014. Financial support from the Japan Society for the Promotion of Science is gratefully acknowledged.

<sup>†</sup>School of Business and Finance, University of International Business and Economics, Beijing 100029, China. Email: rncarpio@yahoo.com

<sup>‡</sup>RIEB, Kobe University, 2-1 Rokkodai, Nada, Kobe 657-8501 Japan. Email: tkamihig@rieb.kobe-u.ac.jp.

# 1 Introduction

It has been known since Bellman and Karush (1962a, 1962b, 1963a, 1963b) that Legendre-Fenchel (LF) duality (Fenchel 1949) can be utilized to solve finite-horizon dynamic programming (DP) problems in discrete time. Although there have been subsequent applications of LF duality to DP (e.g, Morin and Esogbue 1974, Klein 1990, Esogbue and Ahn 1990, Klein and Morin 1991), to our knowledge there has been no serious attempt to exploit LF duality to develop an algorithm to solve infinite-horizon DP problems.

In this paper we propose an algorithm, which we call “Fast Bellman Iteration” (FBI), to compute the value function of an infinite-horizon DP problem in discrete time. FBI is an extremely efficient linear-time algorithm applicable to a class of multidimensional DP problems with concave return functions (or convex cost functions) and linear constraints.

In the FBI algorithm, a sequence of functions is generated starting from the zero function by repeatedly applying a simple algebraic rule involving the LF transform of the return function. The resulting sequence is guaranteed to converge, and the LF transform of the limiting function coincides with the value function. Since the main iteration procedure of FBI involves no optimization, FBI offers a dramatic computational advantage over standard computational methods such as value iteration and policy iteration (e.g., Puterman 2005). We prove that the convergence properties of FBI are identical to those of value iteration when applied to a DP problem with a continuous, bounded, concave return function and a linear constraint.

The rest of the paper is organized as follows. In Section 2 we review some basic concepts from convex analysis and show some preliminary results. In Section 3 we present the general DP framework used in our analysis. In Section 4 we apply LF duality to a DP problem with a continuous, bounded, concave return function and a linear constraint. In Section 5 we present our numerical algorithm and compare its performance with that of modified policy iteration. In Section 6 we offer some concluding comments.

## 2 Preliminaries I: Convex Analysis

In this section we review some basic concepts from convex analysis and state some well-known results. We also establish some preliminary results.

Let  $N \in \mathbb{N}$ . For  $f : \mathbb{R}^N \rightarrow \overline{\mathbb{R}}$ , we define  $f_*, f^* : \mathbb{R}^N \rightarrow \overline{\mathbb{R}}$  by

$$f_*(p) = \inf_{x \in \mathbb{R}^N} \{p^\top x - f(x)\}, \quad \forall p \in \mathbb{R}^N, \quad (1)$$

$$f^*(p) = \sup_{x \in \mathbb{R}^N} \{p^\top x - f(x)\}, \quad \forall p \in \mathbb{R}^N, \quad (2)$$

where  $p$  and  $x$  are  $N \times 1$  vectors, and  $p^\top$  is the transpose of  $p$ . The functions  $f_*$  and  $f^*$  are called the *concave conjugate* and *convex conjugate* of  $f$ , respectively.

It follows from (1) and (2) that for any functions  $f, g : \mathbb{R}_+^N \rightarrow \overline{\mathbb{R}}$ , we have

$$f = -g \quad \Rightarrow \quad \forall p \in \mathbb{R}^N, f_*(p) = -g^*(-p) = -(-f)^*(-p). \quad (3)$$

This allows us to translate any statement about  $g$  and  $g^*$  to the corresponding statement about  $-g$  and  $(-g)_*$ ; this is useful since most results in convex analysis assume convex functions and convex conjugates. In what follows we focus on concave functions and concave conjugates, and by “conjugate,” we always mean “concave conjugate.” The *biconjugate*  $f_{**}$  of  $f$  is defined by

$$f_{**} = (f_*)_* . \quad (4)$$

A concave function  $f : \mathbb{R}^N \rightarrow \overline{\mathbb{R}}$  is called *proper* if  $f(x) < \infty$  for all  $x \in \mathbb{R}^N$  and  $f(x) > -\infty$  for at least one  $x \in \mathbb{R}^N$ . The *effective domain* of  $f$  is defined as

$$\text{dom } f = \{x \in \mathbb{R}^N : f(x) > -\infty\}. \quad (5)$$

Let  $F$  be the set of proper, concave, upper semicontinuous functions from  $\mathbb{R}^N$  to  $\overline{\mathbb{R}}$ . For  $f, g : \mathbb{R}^N \rightarrow \mathbb{R} \cup \{-\infty\}$ , the *sup-convolution* of  $f$  and  $g$  is defined as

$$(f \# g)(x) = \sup_{y \in \mathbb{R}^N} \{f(y) + g(x - y)\}. \quad (6)$$

Some useful known results are collected below:

**Lemma 1** (Rockafellar and Wets 2009, Theorems 11.1, 11.23). (a) *For any  $f \in F$ , we have  $f_* \in F$  and  $f_{**} = f$ .*

(b) *For any  $f, g \in F$ , we have  $(f \# g)_* = f_* + g_*$ .*

(c) Let  $N' \in \mathbb{N}$  and  $u : \mathbb{R}^{N'} \rightarrow \overline{\mathbb{R}}$ . Let  $L$  be an  $N \times N'$  matrix. Define  $(Lu) : \mathbb{R}^N \rightarrow \overline{\mathbb{R}}$  by

$$(Lu)(x) = \sup_{c \in \mathbb{R}^{N'}} \{u(c) : Lc = x\}, \quad \forall x \in \mathbb{R}^N. \quad (7)$$

Then

$$(Lu)_*(p) = u_*(L^\top p), \quad \forall p \in \mathbb{R}^N. \quad (8)$$

The following results are proved in the Appendix.

**Lemma 2.** Let  $A$  be an invertible  $N \times N$  matrix. Let  $\beta \in \mathbb{R}_{++}$  and  $S = A^{-1}/\beta$ . Let  $f, v : \mathbb{R}^N \rightarrow \overline{\mathbb{R}}$  be such that  $f(x) = \beta v(Ax)$  for all  $x \in \mathbb{R}^N$ . Then

$$f_*(p) = \beta v_*(S^\top p). \quad (9)$$

**Lemma 3.** Let  $f, g \in F$  be such that  $\text{dom } f = \text{dom } g = \mathbb{R}_+^N$ . Suppose that both  $f$  and  $g$  are bounded on  $\mathbb{R}_+^N$ . Then  $\text{dom } f^* = \text{dom } g^* = \mathbb{R}_+^N$ , and

$$\sup_{x \in \mathbb{R}_+^N} |f(x) - g(x)| = \sup_{p \in \mathbb{R}_+^N} |f_*(p) - g_*(p)|. \quad (10)$$

### 3 Preliminaries II: Dynamic Programming

In this section we present the general framework for dynamic programming used in our analysis, and show a standard result based on the contraction mapping theorem. Our exposition here is based on Stokey and Lucas (1989) and Kamihigashi (2014a, 2014b).

Let  $N \in \mathbb{N}$  and  $X \subset \mathbb{R}^N$ . Consider the following problem:

$$\sup_{\{x_{t+1}\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t r(x_t, x_{t+1}) \quad (11)$$

$$\text{s.t. } x_0 \in X \text{ given,} \quad (12)$$

$$\forall t \in \mathbb{Z}_+, \quad x_{t+1} \in \Gamma(x_t). \quad (13)$$

In this section we maintain the following assumption.

**Assumption 1.** (i)  $\beta \in (0, 1)$ . (ii)  $\Gamma : X \rightarrow 2^X$  is a nonempty, compact-valued, continuous correspondence. (iii)  $X$  and  $\text{gph } \Gamma$  are convex sets, where  $\text{gph } \Gamma$  is the graph of  $\Gamma$ :

$$\text{gph } \Gamma = \{(x, y) \in \mathbb{R}^N \times \mathbb{R}^N : y \in \Gamma(x)\}. \quad (14)$$

(iv)  $r : \text{gph } \Gamma \rightarrow \mathbb{R}$  is continuous, bounded, and concave.

Let  $\hat{v} : X \rightarrow \mathbb{R}$  be the *value function* of the problem (11)–(13); i.e., for  $x_0 \in X$  we define

$$\hat{v}(x_0) = \sup_{\{x_{t+1}\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t r(x_t, x_{t+1}) \quad \text{s.t. (13)}. \quad (15)$$

It is well-known that  $\hat{v}$  satisfies the optimality equation (see Kamihigashi, 2008, 2014b):

$$\hat{v}(x) = \sup_{y \in \Gamma(x)} \{r(x, y) + \beta \hat{v}(y)\}, \quad \forall x \in X. \quad (16)$$

Thus  $\hat{v}$  is a fixed point of the *Bellman operator*  $B$  defined by

$$(Bv)(x) = \sup_{y \in \Gamma(x)} \{r(x, y) + \beta v(y)\}, \quad \forall x \in X. \quad (17)$$

Let  $\mathcal{C}(X)$  be the space of continuous, bounded, concave functions from  $X$  to  $\mathbb{R}$  equipped with the sup norm  $\|\cdot\|$ . The following result is proved in the Appendix.

**Theorem 1.** *Under Assumption 1, the following statements hold:*

- (a) *The Bellman operator  $B$  is a contraction on  $\mathcal{C}(X)$  with modulus  $\beta$ ; i.e.,  $B$  maps  $\mathcal{C}(X)$  into itself, and for any  $v, w \in \mathcal{C}(X)$  we have*

$$\|Bv - Bw\| \leq \beta \|v - w\|. \quad (18)$$

- (b)  *$B$  has a unique fixed point  $\tilde{v}$  in  $\mathcal{C}(X)$ . Furthermore, for any  $v \in \mathcal{C}(X)$ ,*

$$\forall i \in \mathbb{N}, \quad \|B^i v - \tilde{v}\| \leq \beta^i \|v - \tilde{v}\|. \quad (19)$$

- (c) *We have  $\tilde{v} = \hat{v}$ .*

## 4 The Dual Bellman Operator

In this section we introduce the “dual Bellman operator,” which traces the iterates of the Bellman operator in a dual space for a special case of (11)–(13). In particular we consider the following problem:

$$\max_{\{c_t, x_{t+1}\}_{t=0}^{\infty}} \sum_{t=0}^{\infty} \beta^t u(c_t) \quad (20)$$

$$\text{s.t. } x_0 \in \mathbb{R}_+^N \text{ given,} \quad (21)$$

$$\forall t \in \mathbb{Z}_+, \quad x_{t+1} = Ax_t - Dc_t, \quad (22)$$

$$c_t \in \mathbb{R}_+^{N'}, x_{t+1} \in \mathbb{R}_+^N, \quad (23)$$

where  $A$  is an  $N \times N$  matrix,  $D$  is an  $N \times N'$  matrix with  $N' \in \mathbb{N}$ , and  $c_t$  is a  $N' \times 1$  vector. Throughout this section we maintain the following assumption.

**Assumption 2.** (i)  $\beta \in (0, 1)$ . (ii)  $u : \mathbb{R}_+^{N'} \rightarrow \mathbb{R}$  is continuous, bounded, and concave. (iii)  $A$  is a nonnegative monotone matrix (i.e.,  $Ax \in \mathbb{R}_+^N \Rightarrow x \in \mathbb{R}_+^N$  for any  $x \in \mathbb{R}^N$ ). (iv)  $D$  is a nonzero nonnegative matrix.

It is well-known (e.g., Berman and Plemmons 1994, p. 137) that a square matrix is monotone if and only if it is invertible and its inverse is nonnegative; furthermore, a nonnegative square matrix is monotone if and only if it has exactly one nonzero element in each row and in each column (Kestelman 1973). Thus the latter property is equivalent to part (iii) above.

Under Assumption 2, the optimization problem (20)–(23) is a special case of (11)–(13) with

$$X = \mathbb{R}_+^N, \quad (24)$$

$$\Gamma(x) = \{y \in \mathbb{R}_+^N : \exists c \in \mathbb{R}_+^{N'}, y = Ax - Dc\}, \quad \forall x \in X, \quad (25)$$

$$r(x, y) = \max_{c \in \mathbb{R}_+^{N'}} \{u(c) : y = Ax - Dc\}, \quad \forall (x, y) \in \text{gph } \Gamma. \quad (26)$$

It is easy to see that Assumption 1 holds under Assumption 2 and (24)–(26).

Before proceeding, we introduce a standard convention for extending a function defined on a subset of  $\mathbb{R}^n$  to the entire  $\mathbb{R}^n$ . Given any function  $g : D \rightarrow \overline{\mathbb{R}}$  with  $D \subset \mathbb{R}^n$ , we extend  $g$  to  $\mathbb{R}^n$  by setting

$$g(x) = -\infty, \quad \forall x \in \mathbb{R}^n \setminus D. \quad (27)$$

Note that in general, for any extended real-valued function  $f$  defined on a subset of  $\mathbb{R}^n$ , we have

$$\sup_{x \in \mathbb{R}^n} f(x) = \sup_{x \in \text{dom } f} f(x), \quad (28)$$

where the function  $f$  on the left-hand side is the version of  $f$  extended according to (27).

Letting  $L = A^{-1}D$ , we can express (22) as

$$Lc_t = x_t - A^{-1}x_{t+1}. \quad (29)$$

In view of this and (26), we have

$$r(x, y) = \max_{c \in \mathbb{R}_+^{N'}} \{u(c) : Lc = x - A^{-1}y\}, \quad \forall (x, y) \in \text{gph } \Gamma. \quad (30)$$

By Assumption 2 and (24)–(27), the Bellman operator  $B$  defined by (17) can be written as

$$(Bv)(x) = \sup_{z \in \mathbb{R}^N} \{(Lu)(x - z) + \beta v(Az)\}, \quad \forall x \in X, \quad (31)$$

where  $Lu$  is defined by (7) and (30) with  $z = A^{-1}y$ . The constraint  $y \in \Gamma(x)$  in (17) is implicitly imposed in (31) by the effective domains of  $u$  and  $v$ , which require that there exist  $c \in \mathbb{R}_+^{N'}$  with  $Lc = x - z$  and that  $y = Az \in X$ , respectively. Following the convention (27), we set

$$(Bv)(x) = -\infty, \quad \forall x \in \mathbb{R}^N \setminus X. \quad (32)$$

For any  $f : \mathbb{R}^N \rightarrow \overline{\mathbb{R}}$  with  $\text{dom } f = \mathbb{R}_+^N$ , we write  $f \in \mathcal{C}(X)$  if  $f$  is continuous, bounded, and concave on  $X$ . Since  $u$  is bounded,  $(Bv)(x)$  is well-defined for any  $x \in \mathbb{R}^N$  and  $v : X \rightarrow \overline{\mathbb{R}}$ . In particular, for any  $v \in \mathcal{C}(X)$ , we have  $Bv \in \mathcal{C}(X)$  by Theorem 1. The following result shows that the Bellman operator  $B$  becomes a simple algebraic rule in the “dual” space of conjugates.

**Lemma 4.** *Let  $S = A^{-1}/\beta$ . For any  $v \in \mathcal{C}(X)$  we have*

$$(Bv)_*(p) = u_*(L^\top p) + \beta v_*(S^\top p), \quad \forall p \in \mathbb{R}^N. \quad (33)$$

*Proof.* Let  $f(z) = \beta v(Az)$  for all  $z \in \mathbb{R}^N$ . Let  $g = Lu$ . We claim that

$$Bv = f \# g. \quad (34)$$

It follows from (31) that  $(Bv)(x) = (f \# g)(x)$  for all  $x \in X$ . It remains to show that  $(Bv)(x) = (f \# g)(x)$  for all  $x \in \mathbb{R}^N \setminus X$  or, equivalently,  $(f \# g)(x) > -\infty \Rightarrow x \in X$ . Let  $x \in \mathbb{R}^N$  with  $(f \# g)(x) > -\infty$ . Then there exists  $z \in \mathbb{R}^N$  with  $g(x - z) + f(z) = (Lu)(x - z) + \beta v(Az) > -\infty$ ; i.e.,

$$x - z \in \text{dom}(Lu), \quad Az \in \text{dom } v. \quad (35)$$

Since  $A^{-1}$  and  $D$  are nonnegative by Assumption 2,  $L$  is nonnegative; thus  $\text{dom}(Lu) = \mathbb{R}_+^N$ . Hence  $x - z \geq 0$  and  $Az \geq 0$  by (35). Since the latter inequality implies that  $z \geq 0$  by monotonicity of  $A$ , it follows that  $x \geq z \geq 0$ ; i.e,  $x \in X$ . We have verified (34).

By Lemma 1 we have  $(Bv)_* = (f\#g)_* = f_* + g_*$ . Thus for any  $p \in \mathbb{R}_+^N$ ,

$$(Bv)_*(p) = f_*(p) + g_*(p) = \beta v_*(S^\top p) + u_*(L^\top p), \quad (36)$$

where the second equality uses Lemmas 2 and 1. Now (33) follows.  $\square$

We call the mapping from  $v_*$  to  $(Bv)_*$  defined by (33) the *dual Bellman operator*  $B_*$ ; more precisely, for any  $f : \mathbb{R}^N \rightarrow \overline{\mathbb{R}}$ , we define  $B_*f$  by

$$(B_*f)(p) = u_*(L^\top p) + \beta f(S^\top p), \quad \forall p \in \mathbb{R}^N. \quad (37)$$

Using the dual Bellman operator  $B_*$ , (33) can be written simply as

$$(Bv)_* = B_*v_*, \quad \forall v \in \mathcal{C}(X). \quad (38)$$

According to the convention (27), the domain of a function  $f$  defined on  $\mathbb{R}_+^N$  can always be taken to be the entire  $\mathbb{R}^N$ . Although this results in no ambiguity when we evaluate  $\sup_x f(x)$  (recall (28)), it causes ambiguity when we evaluate  $\|f\|$ . For this reason we specify the definition of  $\|\cdot\|$  as follows:

$$\|f\| = \sup_{x \in \mathbb{R}_+^N} \|f(x)\|. \quad (39)$$

We use this definition of  $\|\cdot\|$  for the rest of the paper. The following result establishes the basic properties of the dual Bellman operator  $B_*$ .

**Theorem 2.** *For any  $v \in \mathcal{C}(X)$ , the following statements hold:*

(a) *For any  $i \in \mathbb{N}$  we have*

$$(B^i v)_* = B_*^i v_*, \quad (40)$$

$$B^i v = (B_*^i v_*)_*, \quad (41)$$

$$\text{dom}(B_*^i v_*) = \mathbb{R}_+^N, \quad (42)$$

where  $B_*^i = (B_*)^i$ .



(b) The sequence  $\{B_*^i v_*\}_{i \in \mathbb{N}}$  converges uniformly to  $\hat{v}_*$  (the conjugate of the value function  $\hat{v}$ ). In particular, for any  $i \in \mathbb{N}$  we have

$$\|B_*^i v_* - \hat{v}_*\| = \|B^i v - \hat{v}\| \leq \beta^i \|v - \hat{v}\| = \beta^i \|v_* - \hat{v}_*\|. \quad (43)$$

(c) We have  $\hat{v} = (\hat{v}_*)_*$ .

*Proof.* (a) We show by induction that (40) holds for all  $i \in \mathbb{N}$ , which implies that  $B_*^i v_* \in \mathcal{C}(X)$ .

Note that for  $i = 1$ , (40) holds by (38). Suppose that (40) holds for  $i = n - 1 \in \mathbb{N}$ :

$$(B^{n-1}v)_* = B_*^{n-1}v_*. \quad (44)$$

Let us consider the case  $i = n$ . We have  $B^{n-1}v \in \mathcal{C}(X)$  by Theorem 1. Thus using (38) and (44), we obtain

$$(B^n v)_* = (BB^{n-1}v)_* = B_*(B^{n-1}v)_* = B_*B_*^{n-1}v_* = B_*^n v_*. \quad (45)$$

Hence (40) holds for  $i = n$ . Now by induction, (40) holds for all  $i \in \mathbb{N}$ .

To see (41), let  $i \in \mathbb{N}$ . Since  $B^i v \in \mathcal{C}(X)$  by Theorem 1, we have  $B^i v = (B^i v)_*$  by Lemma 1. Recalling (4), we have  $B^i v = ((B^i v)_*)_* = (B_*^i v_*)_*$ , where the second equality uses (40). We have verified (41). Finally (42) follows from (41) and Lemma 3 since  $\text{dom}(B^i v) = \mathbb{R}_+^N$ .

(b) By Lemma 3 and (39), for any  $i \in \mathbb{N}$  we have

$$\|B^i v - \hat{v}\| = \|(B^i v)_* - \hat{v}_*\| = \|B_*^i v_* - \hat{v}_*\|, \quad (46)$$

where the second equality uses (40). Thus the first equality in (43) follows; the second equality follows similarly. The inequality in (43) holds by Theorem 1. As a consequence,  $\{B_*^i v_*\}_{i \in \mathbb{N}}$  converges uniformly to  $\hat{v}_*$ .

(c) Since  $\hat{v} \in \mathcal{C}(X)$  by Theorem 1, we have  $\hat{v} = \hat{v}_{**} = (\hat{v}_*)_*$  by Lemma 1. This completes the proof of Theorem 2.  $\square$

Figure 1 summarizes the results of Theorem 2. The vertical bidirectional arrows between  $Bv$  and  $B_*v_*$ ,  $B^2v$  and  $B_*^2v_*$ , etc, indicate that any intermediate result obtained by the Bellman operator  $B$  can be recovered through conjugacy from the corresponding result obtained by the dual Bellman operator  $B_*$ , and vice versa. This is formally expressed by statement (a) of Theorem 2. Statement (b) shows that both iterates  $\{B^i v\}$  and  $\{B_*^i v_*\}$

Figure 1: Bellman operator  $B$  and dual Bellman operator  $B^*$

$$\begin{array}{ccccccc}
 v & \xrightarrow{B} & Bv & \xrightarrow{B} & B^2v & \xrightarrow{B \cdots B} & B^i v & \xrightarrow{i \uparrow \infty} & \hat{v} \\
 \uparrow * & & \uparrow * & & \uparrow * & & \uparrow * & & \uparrow * \\
 v_* & \xrightarrow{B_*} & B_* v_* & \xrightarrow{B_*} & B_*^2 v_* & \xrightarrow{B_* \cdots B_*} & B_*^i v_* & \xrightarrow{i \uparrow \infty} & \hat{v}_*
 \end{array}$$

converge exactly the same way. In fact, as shown by Lemma 3, conjugacy preserves the sup norm between any pair of functions whose effective domains are  $\mathbb{R}_+^N$ . The rightmost vertical arrow in Figure 1 indicates that the value function can be obtained as the conjugate of the limit of  $\{B_*^i v_*\}$ , as shown in statement (c) of Theorem 2.

## 5 Fast Bellman Iteration

We exploit the relations expressed in Figure 1 to construct a numerical algorithm. The upper horizontal arrows in Figure 1 illustrate the standard value iteration algorithm, which approximates the value function  $\hat{v}$  by successively computing  $Bv, B^2v, B^3v, \dots$  until convergence. The same result can be obtained by successively computing  $B_* v_*, B_*^2 v_*, B_*^3 v_*, \dots$  until convergence and by computing the conjugate of the last iterate. Theorem 2(b) suggests that this alternative method achieves convergence essentially in the same number of steps as value iteration, but it is considerably faster since each step is a simple algebraic rule without optimization; recall (37).

Algorithm 1, which we call “Fast Bellman Iteration,” implements this procedure with a finite number of grid points, using nearest-grid-point interpolation to approximate points away from the grid. To be precise, we take  $n$  grid points  $p_1, \dots, p_n$  in  $\mathbb{R}_+^N$  as given, and index them by  $j \in J \equiv \{1, \dots, n\}$ . Recall from (??) that it is unnecessary to consider the behavior of  $B_*^i v_*$  outside  $\mathbb{R}_+^N$ . We also take as given a function  $\rho : \mathbb{R}_+^N \rightarrow \{p_1, \dots, p_n\}$  that maps each point  $p \in \mathbb{R}_+^N$  to a nearest grid point. We define  $\lambda : \mathbb{R}_+^N \rightarrow J$  by  $\rho(p) = p_{\lambda(p)}$ ; i.e.,  $\lambda(p)$  is the index of the grid point corresponding to  $p$ .

Algorithm 1 requires us to compute the conjugate of the function  $u$  at the beginning as well as the conjugate of the final iterate at the end. To compute these conjugates, we employ the linear-time algorithm (linear in the number of grid points) presented in Lucet (1997), which computes the conjugate of

---

**Algorithm 1:** Fast Bellman Iteration

---

let  $n$  grid points in  $\mathbb{R}_+^N$  be given by  $p_1, \dots, p_n \in \mathbb{R}_+^N$ ;  
initialize  $a, b, w : J \rightarrow \mathbb{R}$  (i.e.,  $\forall j \in J, a(j), b(j), w(j) \in \mathbb{R}$ );  
initialize  $g : J \rightarrow J$  (i.e.,  $\forall j \in J, g(j) \in J$ );  
compute  $u_*$  on  $L^\top p_1, \dots, L^\top p_n$ ;  
**for**  $j \leftarrow 1, \dots, n$  **do**  
     $b(j) \leftarrow 0$ ;  
     $w(j) \leftarrow u_*(L^\top p_j) - \mu^\top p_j$ ;  
     $g(j) \leftarrow \lambda(S^\top p_j)$ ;  
fix  $\epsilon > 0$ ;  
 $d \leftarrow 2\epsilon$ ;  
**while**  $d > \epsilon$  **do**  
     $a \leftarrow b$ ;  
    **for**  $j \leftarrow 1, \dots, n$  **do**  
         $b(j) \leftarrow w(j) + \beta a(g(j))$ ;  
     $d \leftarrow \max_{j \in J} \{|a(j) - b(j)|\}$ ;  
compute  $b_*$ ;  
**return**  $b_*$ ;

---

a concave function on a box grid; see Lucet (2010) for a recent survey on computational convex analysis. Since the rate of convergence for  $\{B_*^i v_*\}$  is determined by  $\beta$  (as shown in Theorem 2(b)) and the number of algebraic operations required for each grid point in each iteration of the “while” loop in Algorithm 1 is independent of the number of grid points, it follows that FBI is a linear-time algorithm.

## 5.1 Numerical Comparison

To illustrate the efficiency of FBI, we compare the performance of FBI with that of modified policy iteration (MPI), which is a standard method to accelerate value iteration (Puterman 2005, Ch. 6.5). In what follows, we assume the following in (20)–(23):

$$u(c_1, c_2) = -(c_1 - 10)^2 - (c_2 - 10)^2, \quad \beta = 0.9, \quad D = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \quad (47)$$

Although  $u$  above is not bounded, it is bounded on any bounded region that contains  $L^\top p_1, \dots, L^\top p_n$ ; thus we can treat  $u$  as a bounded function for our purposes. Concerning the matrix  $A$ , we consider two cases:

$$(a) A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad (b) A = \begin{bmatrix} 0 & 1.2 \\ 0.8 & 0 \end{bmatrix}. \quad (48)$$

The grid points for MPI are evenly spread over  $[0, 20] \times [0, 20]$ . The grid points for FBI in the dual space are chosen accordingly.

We implement both FBI and MPI in Python, using the Scipy 0.13.3 package on a 2.40 GHz i7-3630QM Intel CPU. To further accelerate MPI, we utilize C++ to find a policy that maximizes the right-hand side of the Bellman equation (31) by brute-force grid search (we use this approach because a discretized version of a concave function need not be concave; see, e.g., Murota 2003). The resulting policy is used to update the value function 100 times, and then the policy is updated with the new value function.

Table 1 shows the number of iterations and total CPU time for FBI and MPI to converge to a tolerance of  $10^{-5}$ . For each grid size, the final value functions from FBI and MPI are compared by computing, at each grid point, the absolute difference divided by the largest absolute value of the MPI value function; we report the maximum and average values of this difference over all grid points. One can see that the mean difference decreases as the grid size increases; the maximum difference has the same overall tendency.

Panels (a) and (b) in Figure 2 plot the time to convergence of FBI and MPI against the number of grid points using the data in Table 1. Panels (a') and (b') show the performance of FBI for an extended range of grid point sizes. These plots indicate that MPI is a quadratic-time algorithm (since it searches through the entire set of grid points for each grid point), while FBI is a linear-time algorithm, as discussed above. In terms of CPU time, FBI clearly has a dramatic advantage.

Table 1: Number of iterations to convergence, time to convergence in seconds, maximum relative difference, and average relative difference for FBI and MPI algorithms. Case (a) assumes (47) and (48)(a), while case (b) assumes (47) and (48)(b).

	grid size	40x40	80x80	120x120	160x160	200x200	240x240	280x280
FBI	iterations	120	141	178	191	209	203	219
	CPU time	0.045	0.139	0.299	0.587	0.954	1.398	2.086
	iterations	7	7	8	8	8	8	8
(a) MPI	CPU time	1.636	7.964	34.901	93.305	214.673	434.626	764.269
	diff							
diff	max	3.72E-03	3.36E-03	2.31E-03	2.29E-03	1.84E-03	1.79E-03	1.98E-03
	mean	1.20E-03	1.27E-03	8.77E-04	6.91E-04	5.44E-04	7.20E-04	6.34E-04
FBI	iterations	131	193	192	190	205	214	207
	CPU time	0.048	0.184	0.311	0.536	0.943	1.446	1.977
(b) MPI	iterations	8	8	8	8	8	8	8
	CPU time	1.888	9.690	35.887	96.848	220.004	448.523	796.127
diff	max	7.95E-03	5.62E-03	5.01E-03	4.74E-03	4.60E-03	4.52E-03	4.46E-03
	mean	2.63E-03	1.19E-03	1.02E-03	1.40E-03	1.19E-03	1.07E-03	1.19E-03

## 6 Concluding Remarks

In this paper we proposed an algorithm called “Fast Bellman Iteration” (FBI) to compute the value function of an infinite-horizon dynamic programming problem in discrete time. FBI is a linear-time algorithm that offers a dramatic computational advantage for a class of problems with concave return (or convex cost) functions and linear constraints.

The algorithm we presented is based on the theoretical results established for continuous state problems, but in practice, numerical errors are introduced through discretization and computation of conjugates. Although precise error estimates are yet to be established, our numerical experiments suggest that the difference between the value functions computed using FBI and MPI, respectively, are negligible especially when the number of grid points is fairly large. This can also be expected for large-scale applications of FBI.

In concluding the paper we should mention that the theoretical results established in Section 4 can be extended to problems with nonlinear constraints using a general formula of the conjugate of a composite function (Hiriart-Urruty, 2006). An extension to stochastic problems may also be possible due to the linearity of the expectation operator. New algorithms based on such extensions are left for future research.

## A Appendix

### A.1 Proof of Lemma 2

Let  $p \in \mathbb{R}^N$ . Note that

$$f_*(p) = \inf_{x \in \mathbb{R}^N} \{p^\top x - \beta v(Ax)\}. \quad (49)$$

Letting  $y = Ax$  and noticing that  $x = A^{-1}y$ , we have

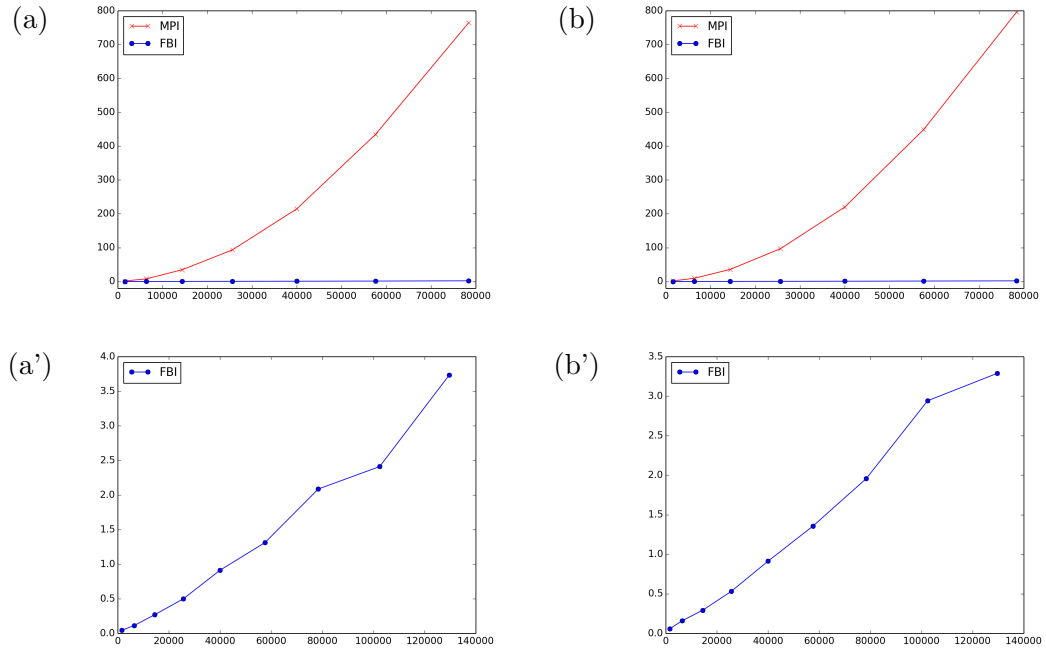
$$f_*(p) = \inf_{y \in \mathbb{R}^N} \{p^\top (A^{-1}y) - \beta v(y)\} \quad (50)$$

$$= \beta \inf_{y \in \mathbb{R}^N} \{(p^\top A^{-1}/\beta)y - v(y)\} \quad (51)$$

$$= \beta \inf_{y \in \mathbb{R}^N} \{(A^{-1}/\beta)^\top p)^\top y - v(y)\}. \quad (52)$$

Now (9) follows.

Figure 2: Time to convergence in seconds vs. number of grid points. Panels (a) & (a') assume (47) and (48)(a), while panels (b) & (b') assume (47) and (48)(b).



## A.2 Proof of Lemma 3

First we establish two lemmas.

**Lemma 5.** Define  $\zeta \in F$  by

$$\zeta(x) = \begin{cases} 0 & \text{if } x \in \mathbb{R}_+^N, \\ -\infty & \text{otherwise.} \end{cases} \quad (53)$$

Then  $\zeta_* = \zeta$ .

*Proof.* For any  $p \in \mathbb{R}^N$  we have

$$\zeta_*(p) = \inf_{x \in \mathbb{R}^N} \{p^\top x - \zeta(x)\} = \inf_{x \in \mathbb{R}_+^N} \{p^\top x - \zeta(x)\} = \inf_{x \in \mathbb{R}_+^N} p^\top x. \quad (54)$$

Thus  $\zeta_*(p) = 0$  if  $p \in \mathbb{R}_+^N$ , and  $\zeta_*(p) = -\infty$  if  $p \in \mathbb{R} \setminus \mathbb{R}_+^N$ ; i.e.,  $\zeta_* = \zeta$ .  $\square$

**Lemma 6.** *Let  $f \in F$  be such that  $\text{dom } f = \mathbb{R}_+^N$  and  $f$  is bounded on  $\text{dom } f$ . Then  $\text{dom } f_* = \mathbb{R}_+^N$  and  $f_*$  is bounded on  $\mathbb{R}_+^N$ .*

*Proof.* Since  $f$  is bounded on  $\text{dom } f$ , there exist  $\underline{\phi}, \bar{\phi} \in \mathbb{R}$  such that  $\underline{\phi} \leq f \leq \bar{\phi}$  on  $\text{dom } f = \mathbb{R}_+^N$ . Let  $p \in \mathbb{R}^N$ . We have

$$f_*(p) = \inf_{x \in \mathbb{R}^N} \{p^\top x - f(x)\} = \inf_{x \in \mathbb{R}_+^N} \{p^\top x - f(x)\}. \quad (55)$$

Hence

$$\inf_{x \in \mathbb{R}_+^N} p^\top x - \bar{\phi} \leq f_*(p) \leq \inf_{x \in \mathbb{R}_+^N} p^\top x - \underline{\phi}. \quad (56)$$

Recalling (54) and Lemma 5, we see that  $\zeta - \bar{\phi} \leq f_* \leq \zeta - \underline{\phi}$ . Thus  $\text{dom } f_* = \text{dom } \zeta = \mathbb{R}_+^N$ , and  $-\bar{\phi} \leq f_* \leq -\underline{\phi}$  on  $\mathbb{R}_+^N$ .  $\square$

To complete the proof of Lemma 3, let  $f, g \in F$  be as in the statement of the lemma. By Lemma 6 we have  $\text{dom } f_* = \text{dom } g_* = \mathbb{R}_+^N$ . Let  $D = \mathbb{R}_+^N$ . It follows from Hiriart-Urruty (1986, p. 484) that

$$\sup_{x \in D} \{f(x) - g(x)\} = \sup_{p \in D} \{g_*(p) - f_*(p)\}, \quad (57)$$

$$\inf_{x \in D} \{f(x) - g(x)\} = \inf_{p \in D} \{g_*(p) - f_*(p)\}. \quad (58)$$

We have

$$\sup_{x \in D} |f(x) - g(x)| = \max\{\sup_{x \in D} \{f(x) - g(x)\}, -\inf_{x \in D} \{f(x) - g(x)\}\} \quad (59)$$

$$= \max\{\sup_{p \in D} \{g_*(p) - f_*(p)\}, -\inf_{p \in D} \{g_*(p) - f_*(p)\}\} \quad (60)$$

$$= \sup_{p \in D} |g_*(p) - f_*(p)|, \quad (61)$$

where the second equality uses (57) and (58). Now (10) follows.

### A.3 Proof of Theorem 1

Let  $C(X)$  be the space of continuous bounded functions from  $X$  to  $\mathbb{R}$  equipped with the sup norm  $\|\cdot\|$ . Then statement (a) holds with  $C(X)$  replacing  $\mathcal{C}(X)$  by Stokey and Lucas (1989, Theorem 4.6). Thus if  $v \in \mathcal{C}(X) \subset C(X)$ , then



$Bv \in C(X)$ ; furthermore,  $Bv$  is concave by a standard argument (e.g., Stokey and Lucas, 1989, p. 81). Thus  $B$  maps  $\mathcal{C}(X)$  into itself. Hence statement (a) holds. It is easy to see that  $\mathcal{C}(X)$  equipped with the sup norm  $\|\cdot\|$  is a complete metric space; thus statement (b) follows by the contraction mapping theorem (Stokey and Lucas, 1989, p. 50). Finally, statement (c) holds by Stokey and Lucas (1989, Theorem 4.3).

## References

- Bellman, R., Karush, W., 1962a, On the maximum transform and semigroups of transformations, *Bulletin of the American Mathematical Society* 68, 516–518.
- Bellman, R., Karush, W., 1962b, Mathematical programming and the maximum transform, *SIAM J. Appl. Math.*, 10, 550–567.
- Bellman, R., Karush, W., 1963a, On the maximum transform, *Journal of Mathematical Analysis and Applications* 6, 67–74.
- Bellman, R., Karush, W., 1963b, Functional equations in the theory of dynamic programming XII: an application of the maximum transform, *Journal of Mathematical Analysis and Applications* 6, 155–157.
- Esogbue, A.O., Ahn, C.W., 1990, Computational experiments with a class of dynamic programming algorithms of higher dimensions, *Computers Math. Applic.* 19, 3–23.
- Fenchel, W., 1949, On conjugate convex functions, *Canadian Journal of Mathematics* 1, 73–77.
- Hiriart-Urruty, J.B., 1986, A general formula on the conjugate of the difference of functions, *Canadian Mathematical Bulletin* 29, 482–485.
- Hiriart-Urruty, J.B., 2006, A note on the Legendre-Fenchel transform of convex composite functions, *Nonsmooth Mechanics and Analysis*, Springer US, 35–46.
- Kamihigashi, T., 2008, On the principle of optimality for nonstationary deterministic dynamic programming, *International Journal of Economic Theory* 4, 519–525.
- Kamihigashi, T., 2014a, An order-theoretic approach to dynamic programming: an exposition, *Economic Theory Bulletin* 2, 13–21.
- Kamihigashi, T., 2014b, Elementary results on solutions to the Bellman equation of dynamic programming: existence, uniqueness, and convergence, *Economic Theory*, 1–23.

- Kestelman, H., 1973, Matrices with  $A \geq 0$  and  $A^{-1} \geq 0$ , *American Mathematical Monthly* E2379 1059–1060.
- Klein, C.M., 1990, Conjugate duality and its implications in dynamic programming, *Math. Comput. Modelling* 14, 151–154.
- Klein, C.M., Morin, T.L., 1991, Conjugate duality and the curse of dimensionality, *European Journal of Operational Research* 50, 220–228.
- Lucet, Y., 1997, Faster than the fast Legendre transform, the linear-time Legendre transform, *Numerical Algorithms* 16, 171–185.
- Lucet, Y., 2010, What shape is your conjugate? A survey of computational convex analysis and its applications, *SIAM Review* 52, 505–542.
- Miller, R.E., Blair, P.D., 2009, *Input-Output Analysis: Foundations and Extensions*, Second Edition. Cambridge University Press, Cambridge.
- Morin, T.L., Esogbue, A.M.O., 1974, The imbedded state space approach to reducing dimensionality in dynamic programs of higher dimensions, *Journal of Mathematical Analysis and Applications* 48, 801–810.
- Murota, K., 2003, *Discrete Convex Analysis*, SIAM, Philadelphia.
- Puterman, M.L., 2005, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Hoboken, New Jersey.
- Rockafellar, R.T., Wets, R.J.B., 2009, *Variational Analysis*, Springer, Dordrecht.
- Stokey, N., Lucas, R.E. Jr., 1989, *Recursive Methods in Economic Dynamics*, Harvard University Press, Cambridge MA.