Dynamic Programming Volumes I and II

A Quick Introduction — Lecture 2 of 2

John Stachurski

Review

In Lecture 1 we saw that DP theory needs to be extended

- no discounting / negative discounting / nonlinear discounting
- state-dependent discounting
- risk preferences / recursive preferences
- ambiguity / desire for robustness / adversarial agents
- distributional DP
- nonstandard timings (Q-factors / structural estimation)
- etc.

This motivates the **abstract dynamic programming** (ADP) framework developed in DP1–2

- A general theory for recursive optimization problems
- Conditions for optimality and convergence of algorithms

Side benefits:

- Separates advanced analysis from foundational DP concepts, proofs
- Facilitates development of new results
- Machine compatible

To abstract and unify, let's recall what's common to the applications we've considered

1. The key primitives are the policy operators T_σ

- Lifetime value v_σ is the fixed point of T_σ for each $\sigma \in \Sigma$
- Value function is defined by $v^* := \sup_{\sigma \in \Sigma} v_\sigma$
- etc.

2. Policy operators are always order preserving

- $v \leqslant w \implies T_\sigma v \leqslant T_\sigma w$

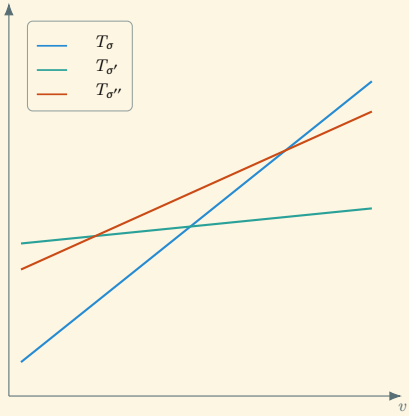
Definition 1

An **abstract dynamic program** (ADP) is a pair $(V, \mathbb{T})$, where

1. $V = (V, \preceq)$ is a partially ordered set and
2. $\mathbb{T} = \{T_\sigma : \sigma \in \Sigma\}$ is a family of order preserving self-maps on V

Here

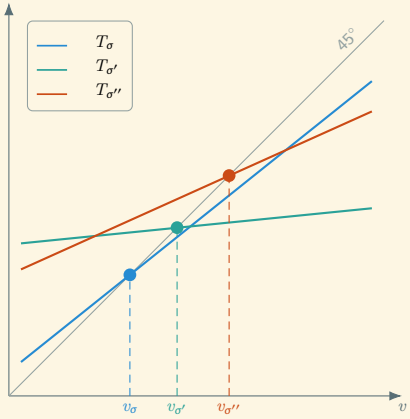
- V is called the **value space**
- Each T_σ is called a **policy operator**
- Each $\sigma \in \Sigma$ is called a **policy**



Eg. A simple ADP with $\Sigma = \{\sigma, \sigma', \sigma''\}$ and $V = \mathbb{R}$

When each T_σ has a unique fixed point v_σ in V, we call the ADP **well-posed**

- v_σ is understood as representing "lifetime value" of σ
- well-posed $\iff$ well-defined objective



The simple ADP is well-posed

All the models we have seen so far are well-posed ADPs

Eg. Inventory model

- Value space V is all $v\colon \mathsf{X} \to \mathbb{R}$ and $\preceq$ is pointwise partial order
- Policy operators take the form

$$(T_\sigma\, v)(x) = \mathbb{E}\left[\pi(x, \sigma(x), D)\right] + \beta\, \mathbb{E}\left[v(h(x, \sigma(x), D))\right]$$

Eg. Distributional DP

- V = all conditional distributions $\eta(x, \cdot)$ with $x \in \mathsf{X}$, order $\preceq$ is pointwise FOSD
- Policy operators take the form $\eta \mapsto D_\sigma \eta$,

$$(D_\sigma\, \eta)(x, B) = \int \left[\int \mathbb{1}_B(r_\sigma(x) + \beta v)\, \eta(x', \mathrm{d}v) \right] P_\sigma(x, \mathrm{d}x')$$

ADP Theory

Let $(V, \mathbb{T})$ be well-posed

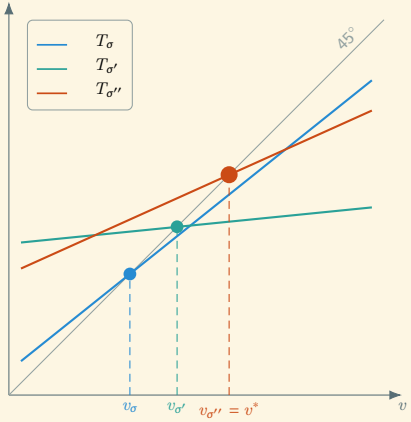
The **value function** is defined as the supremum

$$v^* = \bigvee_{\sigma \in \Sigma} v_\sigma$$

A policy σ is called **optimal** if

$$v_\sigma = v^*$$

Generalizations of the standard definitions

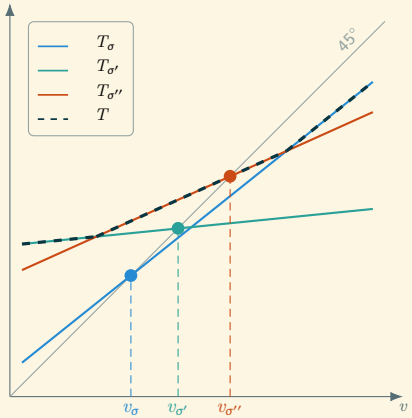


$v^* = v_{\sigma''}$ and σ'' is optimal

What's a systematic way to find optimal policies?

First, we introduce the **Bellman operator** via

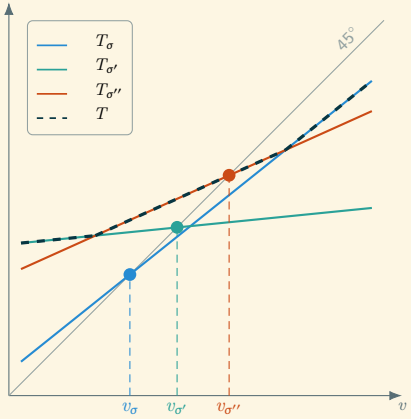
$$Tv = \bigvee_{\sigma \in \Sigma} T_\sigma \, v$$



T is the upper envelope

Next, we say that $v \in V$ satisfies the **Bellman equation** if

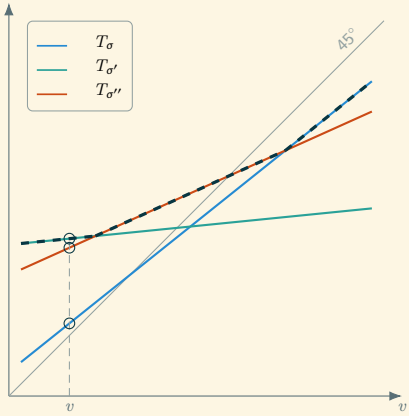
$$v = Tv$$



$v_{\sigma''}$ satisfies the Bellman equation

A policy $\sigma \in \Sigma$ is called v-**greedy** if

$$T_\sigma v = Tv$$



σ' is v-greedy since $T_{\sigma'} v = Tv$

Let $(V, \mathbb{T})$ be a well-posed ADP

Definition 2

We say that the **fundamental optimality properties hold** for $(V, \mathbb{T})$ if

(B1) at least one optimal policy exists

(B2) v^* exists and is the unique solution to the Bellman equation in V_G

(B3) σ is optimal $\iff$ σ is v^*-greedy

- $V_G :=$ all $v \in V$ such that at least one v-greedy policy exists
- Abstract versions of the results we seek in applications...

Lemma 1

For a well-posed ADP, (B2) implies both (B1) and (B3)

Sample Proof: [$v^* \in V_G$ and v^* fixed under T] implies [every v^*-greedy σ is optimal]

- Since $v^* \in V_G$ we can choose a v^*-greedy σ
- For this σ we have $T_\sigma\, v^* = Tv^* = v^*$
- In particular, v^* is a fixed point of T_σ
- By well-posedness, the unique fixed point of T_σ is v_σ
- Hence $v_\sigma = v^*$

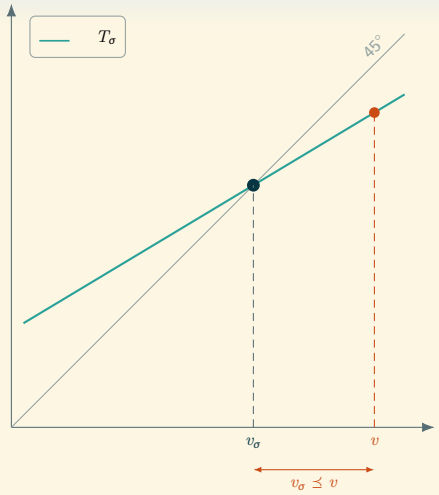
Sufficient Conditions for Optimality Properties

Let $(V, \mathbb{T})$ be a well-posed ADP

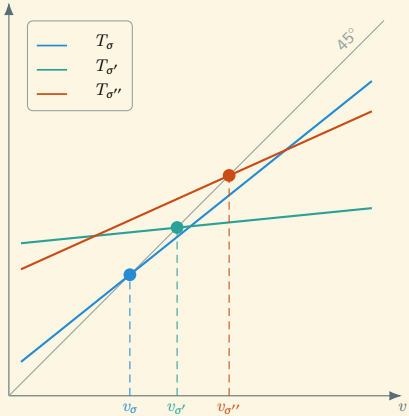
When do the fundamental optimality properties hold?

$(V, \mathbb{T})$ will be called **downward stable** if, for all σ,

$$T_\sigma v \preceq v \implies v_\sigma \preceq v$$



$T_\sigma v \preceq v$ implies $v_\sigma \preceq v$



Eg. The simple three-policy ADP is downward stable

Theorem 1: Max-Optimality

If $(V, \mathbb{T})$ is downward stable, then the following statements are equivalent:

(i) T has a fixed point and that fixed point admits a greedy policy

(ii) The fundamental optimality properties hold

Proof: See DP2

Applications

Let's revisit some of the problems we looked at earlier

We wish to verify that the fundamental optimality properties hold

Example: State-Dependent Discounting

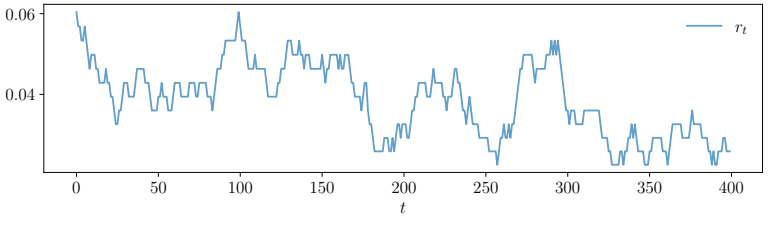
Policy operators

$$(T_\sigma v)(x, z) = \mathbb{E}\left[\pi(x, \sigma(x), D) + \beta(z) \int v(h(x, \sigma(x), D), z')\, Q(z, \mathrm{d}z')\right]$$

Under some continuity and "eventual" discounting conditions (see DP2),

- downward stability holds
- T has a fixed point that admits a greedy policy

Hence fundamental optimality properties hold


0.06
0.04
r_t
0
50
100
150
200
250
300
350
400
t
20
15
10
5
0
inventory
0
50
100
150
200
250
300
350
400
t

Example: Risk-Adjusted Expectations

Recall the inventory problem with policy operators

$$(T_\sigma\, v)(x) = \mathscr{E}\left[\pi(x, \sigma(x), D) + \beta\, v(h(x, \sigma(x), D))\right]$$

Under some conditions on $\mathscr{E}$ (see DP2),

- downward stability holds and
- T has a fixed point that admits a greedy policy

Hence fundamental optimality properties hold

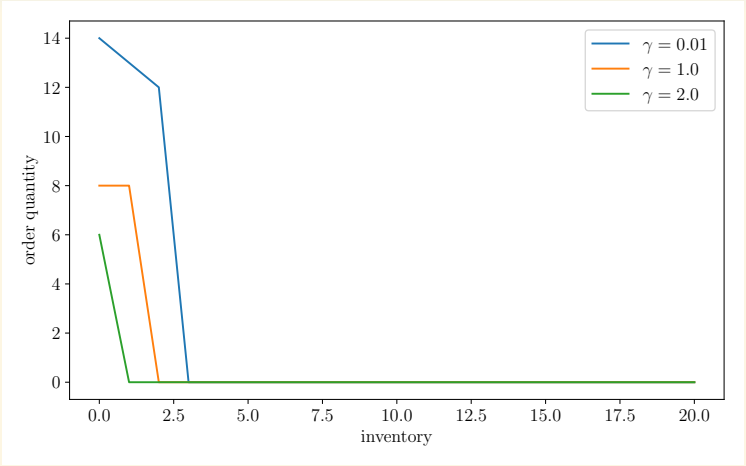


Figure 1: Risk-sensitive firms order less aggressively

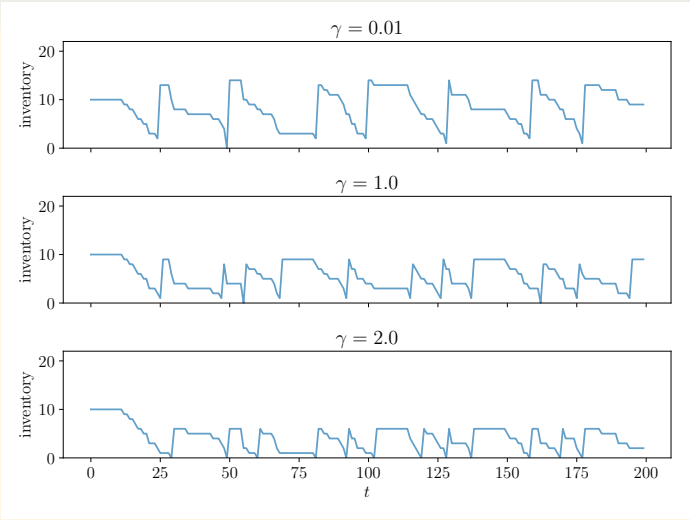


Figure 2: Inventory dynamics: more risk-sensitive firms hold less stock

Example: Model Uncertainty

Recall the investment problem with model uncertainty

$$v(k,z) = \max_{0 \leqslant i} \left\{ f(k,z) - i + \beta \inf_{\psi \ll \varphi} \left[\int v(k', g(z,x)) \psi(\mathrm{d}x) + \frac{1}{\gamma} D(\psi \,\|\, \varphi) \right] \right\}$$

Under some conditions on D (see DP2),

1. downward stability holds and
2. T has a fixed point that admits a greedy policy

As a result, the fundamental optimality properties hold

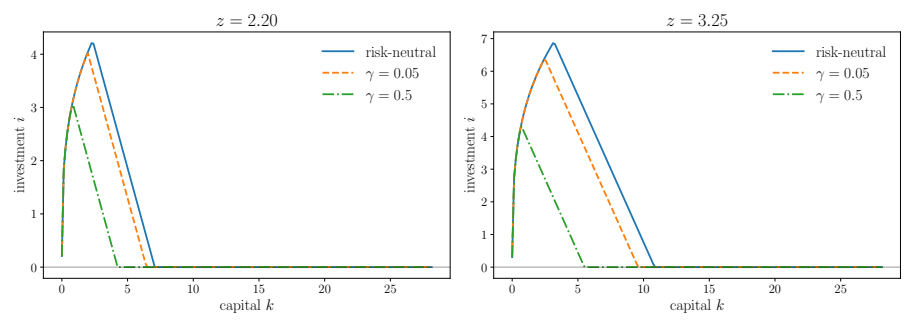


Figure 3: Investment policies

Algorithms

Recall the big three algorithms: **VFI, OPI, and HPI**

Abstract versions:

- **VFI:** iterate with $T\colon\ v \mapsto Tv$
- **OPI:** iterate with $W\colon\ v \mapsto T_\sigma^m v$ where σ is v-greedy
- **HPI:** iterate with $H\colon\ v \mapsto v_\sigma$ where σ is v-greedy

VFI: Iterating with T

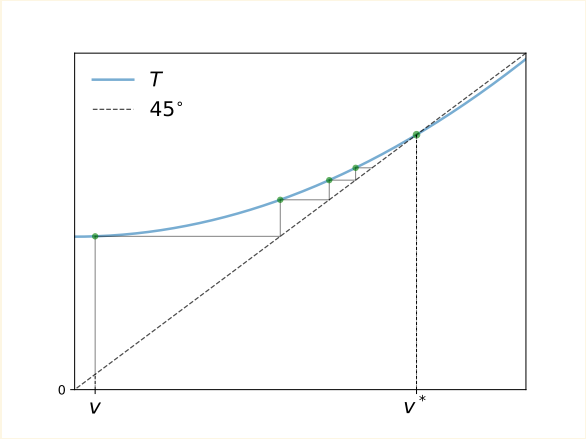


Figure 4: VFI: iterating with T

HPI: Iterating with H

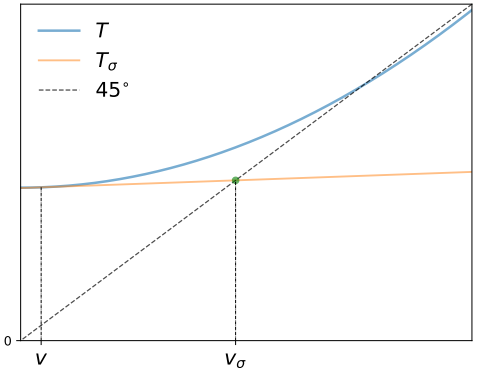


Figure 5: Choosing σ to be v-greedy implies $T_\sigma v = Tv$

HPI: Two Steps

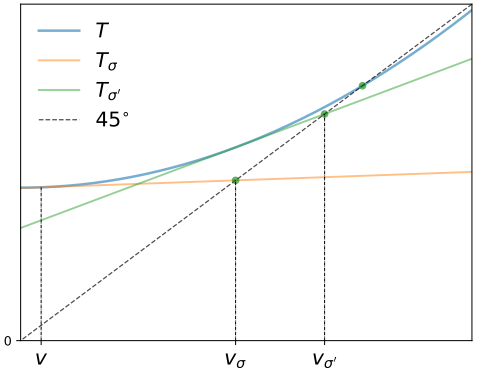


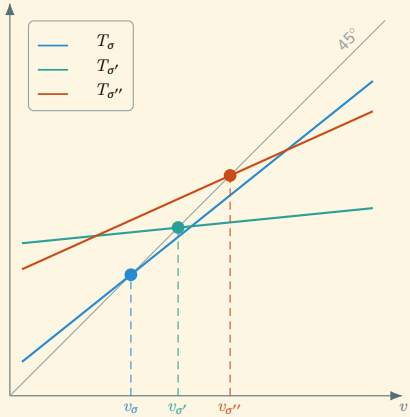
Figure 6: Two HPI steps: $v \to v_\sigma \to v_{\sigma'}$

Convergence Results for Algorithms

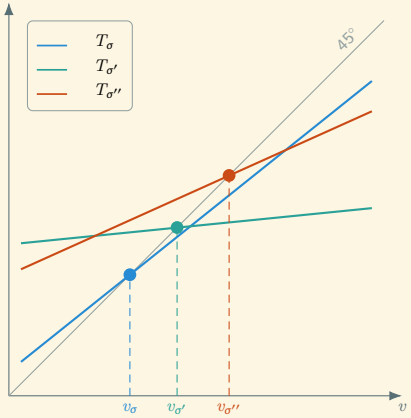
Suppose that V has a topology as well as a (closed) partial order

We say that $(V, \mathbb{T})$ is **globally stable** if each $T_\sigma \in \mathbb{T}$ is globally stable:

$$T_\sigma \text{ has a unique fixed point } v_\sigma \in V \text{ and } T_\sigma^n v \to v_\sigma \text{ for all } v \in V$$



Eg. The simple three-policy ADP is globally stable



The simple ADP is globally stable

Theorem 2: Max-Optimality under Global Stability

Let $(V, \mathbb{T})$ be globally stable. If $V_G = V$ and T has a fixed point in V, then

1. the fundamental optimality properties hold
2. VFI, OPI and HPI all converge

Algorithms: Relationships

Proposition 1

Let the conditions of the previous theorem hold.

In this setting, for all $n \in \mathbb{N}$ and v with $v \preceq Tv$,

$$v \preceq T^n v \preceq W^n v \preceq H^n v \preceq v^*$$

Remarks:

- HPI is "faster" than OPI, which is "faster" than VFI
- Proofs are simplified — VFI converges $\implies$ all converge

Aside: Which is actually faster?

It depends on the problem but also on changing software/hardware

VFI involves a large number of small steps (more sequential)

HPI involves a small number of large steps (more potential for parallelization)

OPI offers flexibility by varying m

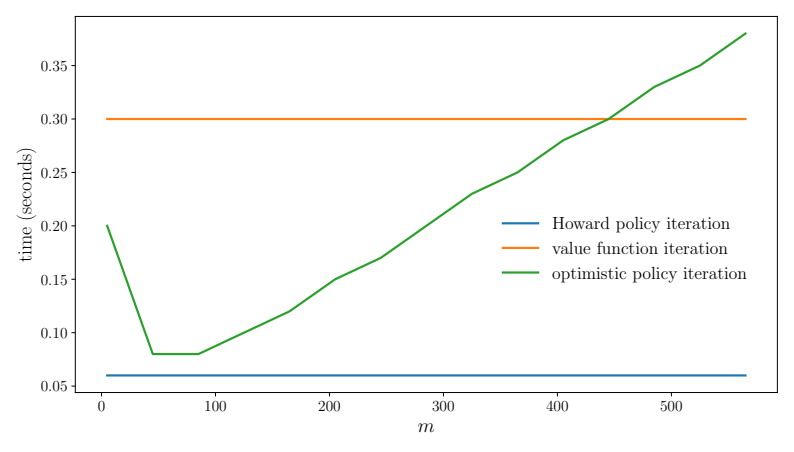


Figure 7: Solver times (JAX on GPU, finite states and actions)

Minimization

Definitions

- **Min-value function**: $v_\nabla^* \coloneqq \bigwedge_\sigma v_\sigma$
- **Min-Bellman operator**: $T_\nabla v \coloneqq \bigwedge_{\sigma \in \Sigma} T_\sigma v$
- σ is called v-**min-greedy** if $T_\sigma v = T_\nabla v$
- σ is called **min-optimal** if $v_\sigma = v_\nabla^*$

We obtain mirror images of our previous max-optimality results:

Theorem 3: Min-Optimality

If $(V, \mathbb{T})$ is upward stable, then the following statements are equivalent:

(i) T_∇ has a fixed point in V_∇^G

(ii) The fundamental min-optimality properties hold

Theorem 4: Min-Optimality and Convergence

Let $(V, \mathbb{T})$ be globally stable. If $V_\nabla^G = V$ and T_∇ has a fixed point in V, then

1. the fundamental min-optimality properties hold and
2. min-VFI, min-OPI and min-HPI all converge

Proofs

How to avoid re-doing proofs for the min case?

Key idea: infima in $(V, \preceq)$ correspond to suprema in the **order-dual** $V^\partial := (V, \preceq^\partial)$, where order is reversed

The **order-dual** of ADP $(V, \mathbb{T})$ is $(V^\partial, \mathbb{T})$

Steps

1. State hypotheses for min result in $(V, \mathbb{T})$
2. Infer corresponding properties in order dual $(V^\partial, \mathbb{T})$
3. Solve max problem in order-dual using previous max results
4. Shift results back to $(V, \mathbb{T})$

Example: Negative Discount Optimality

Recall the negative discounting problem with Bellman min-equation

$$v(x) = \min_{0 \leqslant a \leqslant x} \{c(a) + \delta\, v(x-a)\}$$

- $c(0) = 0,\ c' > 0,\ c'' > 0,\ \delta > 1$

Policy operators:

$$(T_\sigma v)(x) = c(\sigma(x)) + \delta\, v(x - \sigma(x))$$

Acting on V = all increasing $v\colon [0,1] \to \mathbb{R}$ with $v(0) = 0$

Since $\delta > 1$, these operators are not contractions

However, we can show that the fundamental min-optimality properties hold

Steps:

1. Pick a suitably restricted class of policies Σ
2. Show that T_σ is upward stable on V for each σ in Σ
3. Show that T_∇ has a fixed point in $v \in V$
4. Show that this fixed point has a v-min-greedy policy

Now apply Theorem 3

One consequence: We can construct an equilibrium for the production chain model from Lecture 1, where equilibrium prices obey

$$p(s) = \min_{0 \leqslant t \leqslant s} \{c(s-t) + \delta\, p(t)\}$$

(Value function of negative DP problem is an equilibrium price function)

FOC:

$$c'(s-t) = \delta\, p'(t)$$

Coase (1937): "a firm will tend to expand until the costs of organizing an extra transaction within the firm become equal to the costs of carrying out the same transaction by means of an exchange on the open market."

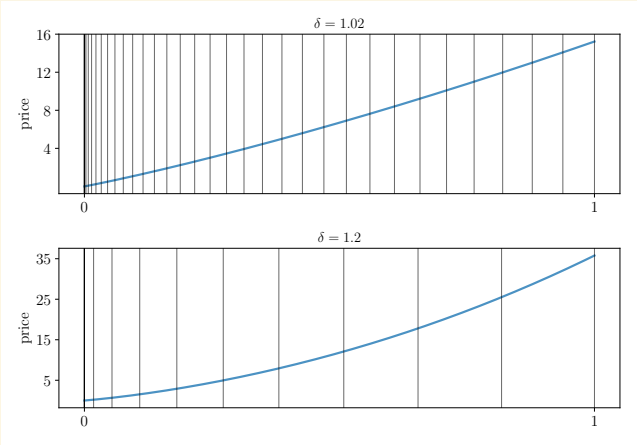


Figure 8: Equilibrium prices and firm boundaries for two values of δ

Example: Sequential Analysis

The problem can be reduced to an optimal stopping problem with Bellman min-operator

$$(T_\triangledown\, g)(\pi) = \min\left\{e(\pi),\ c + \int g(\pi')\, P(\pi, \mathrm{d}\pi')\right\}$$

Can show:

- Globally stable
- $T_\triangledown$ has a fixed point in V

Hence fundamental min-optimality properties hold and min-algorithms converge

Factorization

Some interesting ADPs come in pairs

Eg. In structural estimation the **post-action value function** g satisfies

$$g(x, a) = \int \max_{a' \in \mathsf{A}} [r(x', a') + \beta g(x', a')] \, P(x, a, \mathrm{d}x')$$

This is obviously related to the traditional Bellman equation

$$v(x) = \max_{a \in \mathsf{A}} \left\{ r(x, a) + \beta \int v(x') P(x, a, \mathrm{d}x') \right\}$$

What is the exact relationship between them?

Eg. MDP model with Bellman equation

$$v(x) = \max_{a \in \Gamma(x)} \left\{ r(x, a) + \beta \int v(x') P(x, a, \mathrm{d}x') \right\}$$

The **Q-factor representation** (used for Q-learning) has Bellman equation

$$q(x, a) = r(x, a) + \beta \int \max_{a' \in \Gamma(x')} q(x', a') \, P(x, a, \mathrm{d}x')$$

What is the exact relationship between them?

Eg. Risk-sensitive inventory problem with Bellman equation

$$v(x) = \max_{a \in \Gamma(x)} \psi^{-1} \left\{ \mathbb{E}\, \psi \left[\pi(x, a, D) + \beta v(h(x, a, D)) \right] \right\},$$

- Entropic certainty equivalent: $\psi(t) = \exp(-\gamma t)$ with $\gamma > 0$

A guess of the corresponding Q-factor Bellman equation:

$$q(x, a) = \mathbb{E}\left[\exp(-\gamma\, \pi(x, a, D)) \cdot \left(\min_{a' \in \Gamma(x')} q(x', a') \right)^{\beta} \right]$$

where $x' \coloneqq h(x, a, D)$

- Is this correct?
- Do these problems have the same optimal policies?
- Does Q-learning work here?

More generally

- How can we transform dynamic programs systematically?
- When are the fundamental optimality properties preserved?
- How are the optimal policies related?
- When is convergence of algorithms preserved?
- How do I implement these algorithms?

Factored Dynamic Programs

The key idea: factor each policy operator:

Eg. For a finite MDP with

$$(T_\sigma v)(x) = r(x, \sigma(x)) + \beta \int v(x')P(x, \sigma(x), \mathrm{d}x')$$

we can write $T_\sigma = G_\sigma \circ F$ where

- $F\colon v(x) \mapsto r(x, a) + \beta \int v(x')P(x, a, \mathrm{d}x')$
- $G_\sigma\colon g(x, a) \mapsto g(x, \sigma(x))$

Now consider iterating with $\hat{T}_\sigma = F \circ G_\sigma$ instead...

Since

- $F\colon v(x) \mapsto r(x,a) + \beta \int v(x')P(x,a,\mathrm{d}x')$
- $G_\sigma\colon g(x,a) \mapsto g(x,\sigma(x))$

setting $\hat{T}_\sigma = F \circ G_\sigma$ yields

$$(\hat{T}_\sigma q)(x,a) = r(x,a) + \beta \int q(x',\sigma(x'))\, P(x,a,\mathrm{d}x')$$

The Bellman equation for this new ADP is

$$q(x,a) = r(x,a) + \beta \int \max_{a' \in \Gamma(x')} q(x',a')\, P(x,a,\mathrm{d}x')$$

This is the Q-factor Bellman equation, for Q-learning!

In general, there are multiple ways that we can factor T_σ

- We'll see more examples below

And for complex ADPs, there can be a very large number of possible factorizations

This means that we have many ways to "rearrange" ADPs in possibly advantageous ways

What we want is

- Systematic ways to explore this
- Systematic ways to understand the relationships between such ADPs

FDPs

A **factored dynamic program** (FDP) is a tuple $(V, F, \hat{V}, \mathbb{G})$

- V and $\hat{V}$ are nonempty posets
- $F\colon V \to \hat{V}$ and $G_\sigma\colon \hat{V} \to V$
- $\{G_\sigma \hat{v}\}_{\sigma\in\Sigma}$ has a greatest element for every $\hat{v} \in \hat{V}$

Each FDP generates two ADPs

- **Primary ADP** $(V, \mathbb{T})$: $T_\sigma = G_\sigma \circ F$
- **Subordinate ADP** $(\hat{V}, \hat{\mathbb{T}})$: $\hat{T}_\sigma = F \circ G_\sigma$

Example: Structural Estimation

The policy operators for the original MDP model take the form

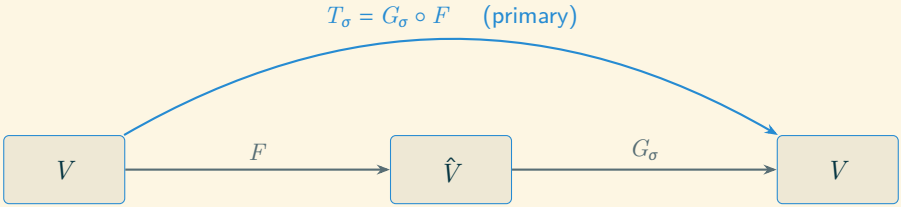
$$(T_\sigma v)(x) = r(x, \sigma(x)) + \beta \int v(x') P(x, \sigma(x), \mathrm{d}x').$$

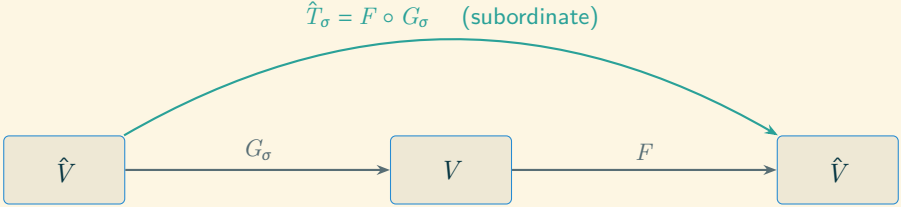
The post-action value function policy operators take the form

$$(\hat{T}_\sigma g)(x, a) = \int \left[r(x', \sigma(x')) + \beta g(x', \sigma(x')) \right] P(x, a, \mathrm{d}x')$$

These are primary and subordinate for the FDP with

- $F\colon v(x) \mapsto \int v(x') P(x, a, \mathrm{d}x')$ (state $\longrightarrow$ state-action)
- $G_\sigma\colon g(x, a) \mapsto r(x, \sigma(x)) + \beta\, g(x, \sigma(x))$ (state-action $\longrightarrow$ state)


$T_\sigma = G_\sigma \circ F$ (primary)
V
F
$\hat{V}$
G_σ
V


$\hat{T}_\sigma = F \circ G_\sigma$ (subordinate)
$\hat{V}$
G_σ
V
F
$\hat{V}$

FDP Max-Optimality

Optimality can be studied via the subordinate ADP

Theorem 5: FDP Optimality (Order Preserving Case)

Let F and G_σ be order preserving

In this setting,

1. The fundamental optimality properties hold for the primary ADP if and only if they hold for the subordinate ADP
2. Any optimal policy for the primary ADP is optimal for the subordinate ADP
3. If $\hat{v}^*$ is the value function of the subordinate ADP and $G_\sigma \hat{v}^* = G\hat{v}^*$, then σ is optimal for the primary ADP

FDP Min-Optimality

Theorem 6: FDP Optimality (Order Reversing Case)

Let F and G_σ be order reversing

In this setting,

1. The max-optimality properties hold for the primary ADP if and only if the min-versions hold for the subordinate ADP
2. Any max-optimal policy for the primary ADP is min-optimal for the subordinate ADP
3. If $\hat{v}^*_\nabla$ is the min-value function of the subordinate ADP and $G_\sigma \hat{v}^*_\nabla = G\hat{v}^*_\nabla$, then σ is max-optimal for the primary ADP

Eg. To solve for an optimal policy for the original ADP

$$v(x) = \max_a \left\{ r(x, a) + \beta \int v(x') P(x, a, \mathrm{d}x') \right\}$$

we can solve the structural estimation Bellman equation

$$g(x, a) = \int \max_{a'} \left[r(x', a') + \beta g(x', a') \right] P(x, a, \mathrm{d}x')$$

to produce the fixed point g^* and then compute σ via

$$\sigma(x) \in \arg\max_a \{ r(x, a) + \beta \, g^*(x, a) \}$$

This is a relatively easy case — we'll see a more complex one in the next section

Reinforcement Learning

Consider a finite MDP with Bellman equation

$$v(x) = \max_{a \in \Gamma(x)} \left\{ r(x, a) + \beta \sum_{x'} v(x') P(x, a, x') \right\}$$

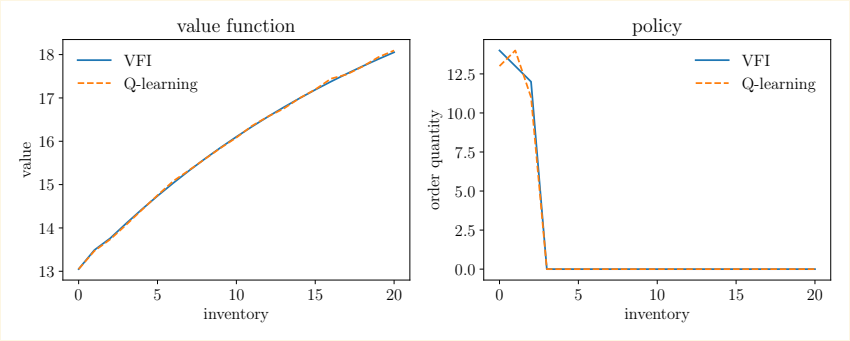
Reinforcement learning (RL) solves for optimal policies without knowledge of P, r

Learn the optimal Q-factor: update via

$$q_{t+1}(X_t, A_t) = (1 - \alpha_t)\, q_t(X_t, A_t) + \alpha_t \left[R_t + \beta \max_{a'} q_t(X_{t+1}, a') \right]$$

- Converges to the Q-factor fixed point q^* under standard conditions

Example: Inventory problem (risk neutral)



Risk-Sensitive Q-Factor

Consider the risk-sensitive inventory policy operator

$$(T_\sigma v)(x) = \psi^{-1}\{\mathbb{E}\,\psi\,[\pi(x, \sigma(x), D) + \beta\, v(h(x, \sigma(x), D))]\}$$

This is the primary ADP when

$$(Fv)(x, a) := \mathbb{E}\exp[-\gamma\,(\pi(x, a, D) + \beta\; v(h(x, a, D)))]$$

and

$$(G_\sigma\; q)(x) := -\frac{1}{\gamma}\ln\big(q(x, \sigma(x))\big)$$

Since $\gamma > 0$, both F and G_σ are order reversing:

$$(Fv)(x, a) \coloneqq \mathbb{E} \exp[-\gamma\,(\pi(x, a, D) + \beta\, v(h(x, a, D)))]$$

and

$$(G_\sigma\, q)(x) \coloneqq -\frac{1}{\gamma} \ln\big(q(x, \sigma(x))\big)$$

The subordinate ADP has policy operators

$$(\hat{T}_\sigma q)(x, a) = \mathbb{E}\left[\exp(-\gamma\, \pi(x, a, D)) \cdot\; q(h(x, a, D), \sigma(h(x, a, D)))^\beta\right]$$

We can use it to recover the optimal policy for the original problem

Risk-Sensitive Q-Learning

The Q-factor Bellman equation corresponding to the subordinate ADP is

$$q(x,a) = \mathbb{E}\left[\exp(-\gamma\,\pi(x,a,D)) \cdot \left(\min_{a' \in \Gamma(x')} q(x',a')\right)^{\beta}\right]$$

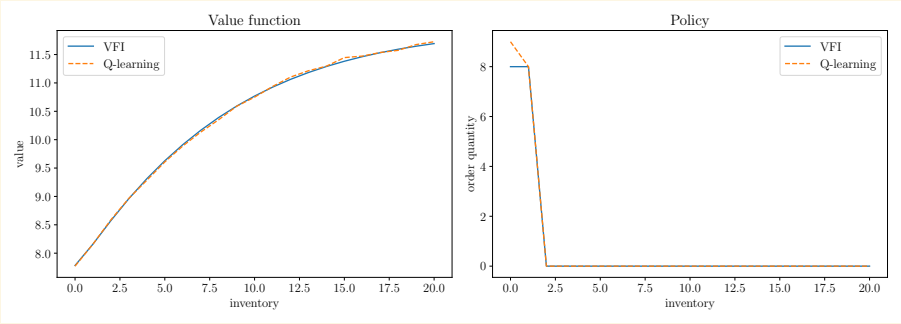
where $x' = h(x,a,D)$ — confirms our earlier guess

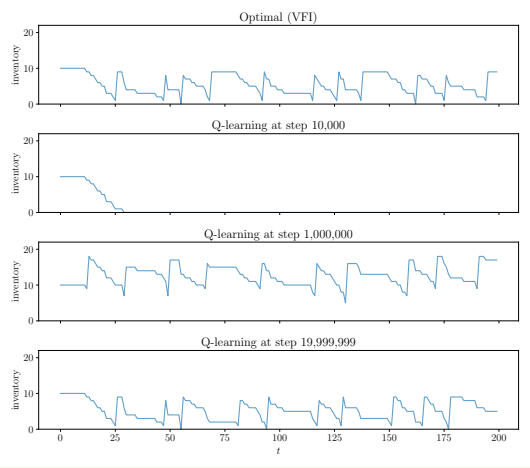
We can learn this risk-sensitive Q-factor from data

The Q-learning update rule becomes

$$q_{t+1}(x,a) = (1-\alpha_t)\,q_t(x,a) + \alpha_t\left[\exp(-\gamma\,R_{t+1}) \cdot \left(\min_{a'} q_t(X_{t+1},a')\right)^{\beta}\right]$$

Risk-sensitive Q-learning converges




Optimal (VFI)
Q-learning at step 10,000
Q-learning at step 1,000,000
Q-learning at step 19,999,999
inventory
0
10
20
0
25
50
75
100
125
150
175
200
t

Conclusion

DP1–DP2 develop an abstract framework for DP

- ADPs: families of order-preserving policy operators on a poset
- Conditions for fundamental optimality properties
- Conditions for convergence of VFI, OPI, and HPI
- Order-duality transfers max results to the min case
- Factorization relates ADPs and underpins Q-learning extensions

The framework covers many nonstandard cases

- state-dependent, negative, and nonlinear discounting
- risk-sensitive and ambiguity-averse preferences
- sequential analysis and stochastic shortest path
- structural estimation and distributional DP
- risk-sensitive Q-learning

Proofs are short and algebraic — a natural target for machine-assisted reasoning and formalization

References

Coase, R. H. (1937). The nature of the firm. *Economica*, 4(16):386–405.