Why Should I Trust You, Bellman? The Bellman Error is a Poor Replacement for Value Error

Scott Fujimoto, David Meger, Doina Precup, Ofir Nachum, Shane Gu

Overview of Paper

We show the Bellman error is a poor proxy for value error.

Reasons:

(1) The magnitude of the Bellman error hides bias.

(2) The Bellman equation has infinite solutions over an incomplete dataset.

Bellman Error & Value Error

Given an approximate value function Q.

Bellman Error

$$\epsilon(s,a) \coloneqq Q(s,a) - E[r + \gamma Q(s',a')]$$

= The difference of each side of the Bellman equation.

Value Error

$$\Delta(s,a) \coloneqq Q(s,a) - Q^{\pi}(s,a)$$

= The difference between Q and the true value function.

Bellman Error & Value Error

Value error is our actual objective, but is typical unavailable.

Known result:

If the Bellman error = 0 for all state-action pairs, then value error = 0.

This suggests that the Bellman error can be used as a proxy.

Problem 1: The Magnitude of the Bellman Error Hides Bias

From the Bellman equation:

$$\epsilon(s,a) = \Delta(s,a) - \gamma E_{s',a'\sim\pi}[\Delta(s',a')]$$

This means that $\Delta(s,a)$ can cancel with $\Delta(s',a')$.

$\Rightarrow$ Biased value functions will have lower Bellman error.

Example

Given the true value function Q^π for any MDP.

Define, for all (s,a):

$$Q_1(s,a) \coloneqq Q^\pi(s,a) + 1$$

$$Q_2(s,a) \coloneqq Q^\pi(s,a) \pm 1$$

Where $\pm$ is random with equal probability of being 1 or -1.

The absolute value error of Q_1 and Q_2 at any (s,a) is 1.

Example

Recall

$$\epsilon(s,a) = \Delta(s,a) - \gamma E_{s',a' \sim \pi}[\Delta(s',a')]$$

For all (s,a) …

Bellman error of $Q_1 := Q^\pi + 1$

$= 1 - \gamma 1 = 1 - \gamma$

Expected Bellman error of $Q_2 := Q^\pi \pm 1$

$= E[|\pm 1 - \gamma E[\pm 1]|] = E|\pm 1 - 0| = 1$

Example

Recall

$\epsilon(s,a) = \Delta(s,a) - \gamma E_{s',a' \sim \pi}[\Delta(s',a')]$

For all (s,a) …

Bellman error of $Q_1 := Q^\pi + 1$

$= 1 - \gamma 1 = 1 - \gamma$

Expected Bellman error of $Q_2 := Q^\pi \pm 1$

$= E[|\pm 1 - \gamma E[\pm 1]|] = E|\pm 1 - 0| = 1$

Example

Recall

$\epsilon(s,a) = \Delta(s,a) - \gamma E_{s',a'\sim\pi}[\Delta(s',a')]$

For all (s,a) ...

Bellman error of $Q_1 := Q^\pi + 1$

$= 1 - \gamma 1 = 1 - \gamma$

Expected Bellman error of $Q_2 := Q^\pi \pm 1$

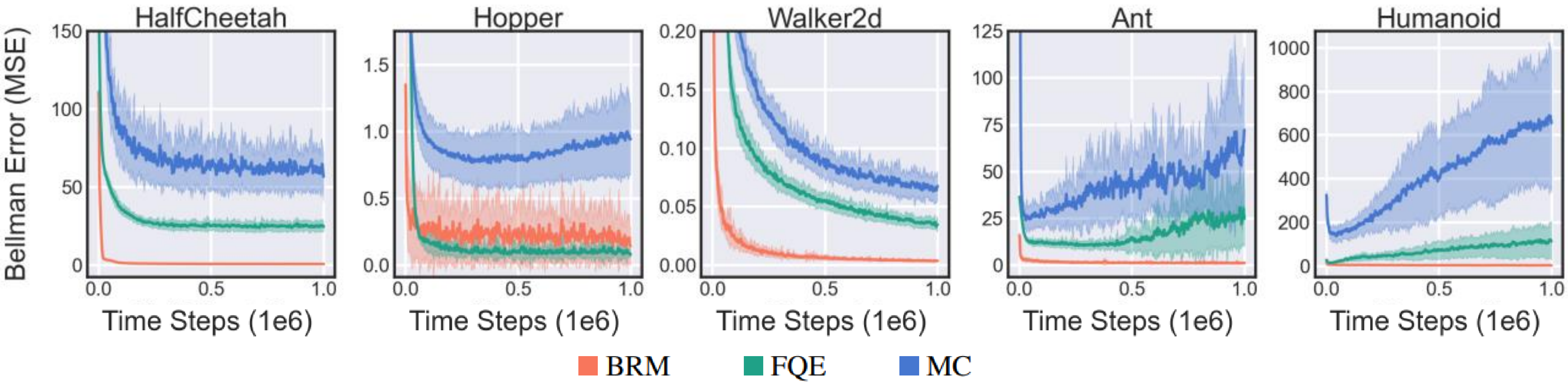
$= E[|\pm 1 - \gamma E[\pm 1]|] = E|\pm 1 - 0| = 1$

Experiment

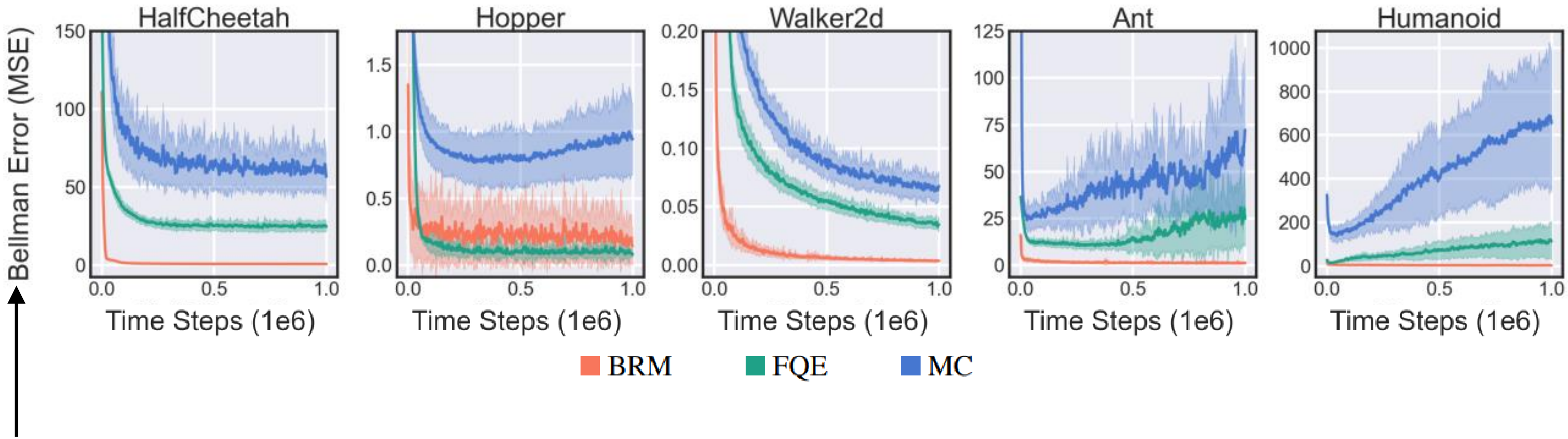
Setup:

On-policy evaluation from large dataset (1m samples).

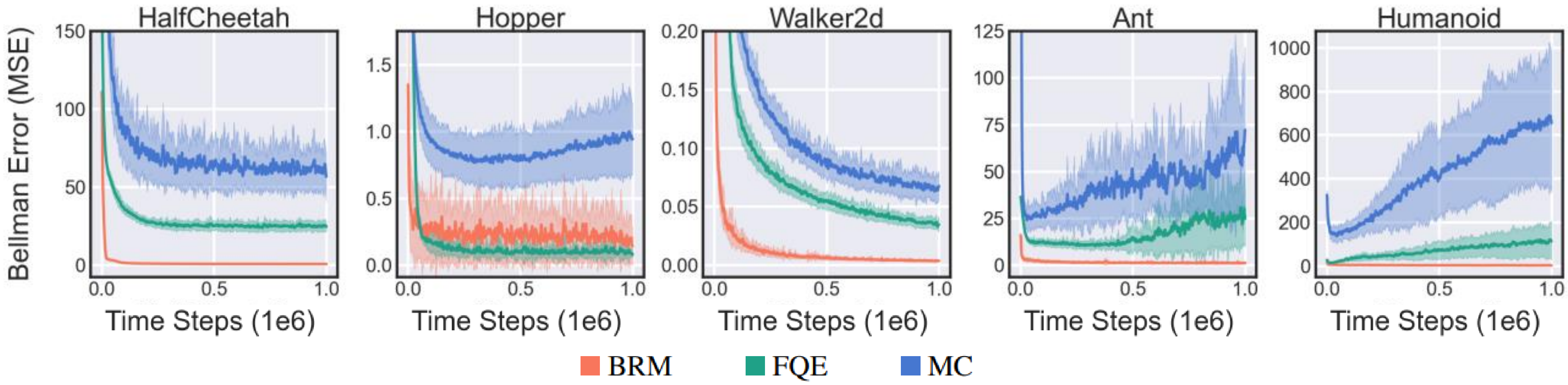
Deterministic environment = no double-sampling issue.



HalfCheetah
Hopper
Walker2d
Ant
Humanoid
Bellman Error (MSE)
Time Steps (1e6)
BRM
FQE
MC



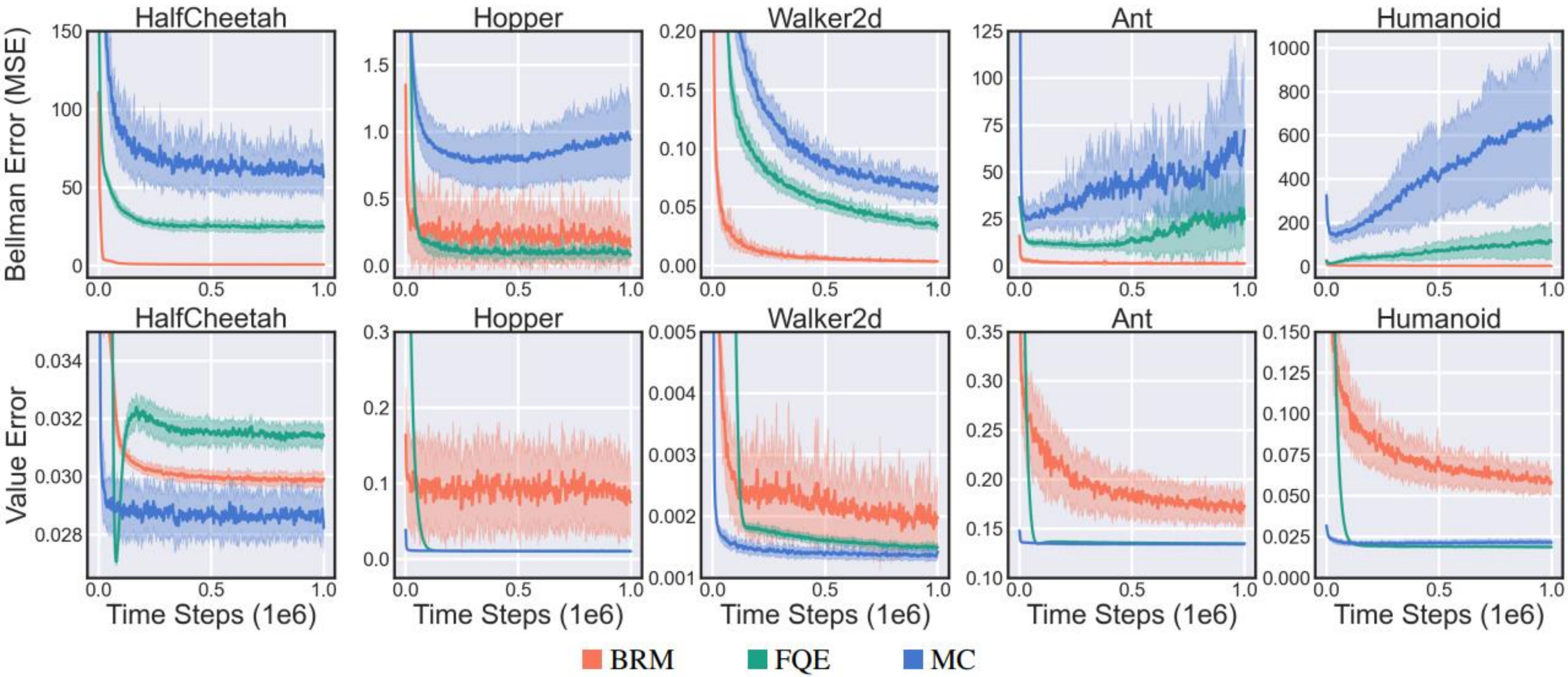
Bellman Error

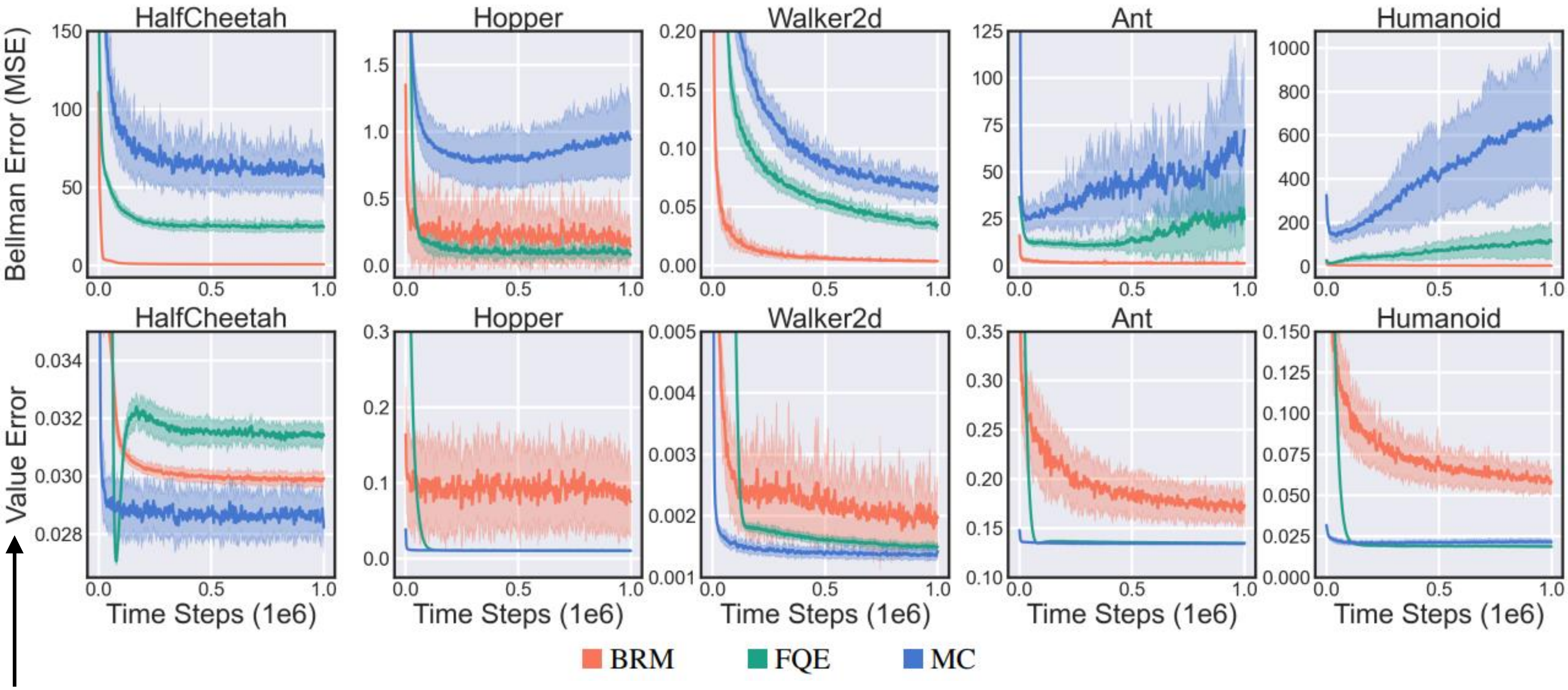


Monte-Carlo Policy Evaluation (MC) does not consider the Bellman error.

Fitted Q-Evaluation (FQE) indirectly minimize the Bellman error.

Bellman Residual Minimization (BRM) directly minimizes the Bellman error.


HalfCheetah
Hopper
Walker2d
Ant
Humanoid
Bellman Error (MSE)
Value Error
Time Steps (1e6)
BRM
FQE
MC



Normalized Value Error

Problem 2: Missing Transitions Breaks the Bellman Equation

Can we make Bellman error work as an off-policy objective?

Recall:

If the Bellman error = 0 for all state-action pairs, then value error = 0.

What if we are missing data?

Bellman Equations over an Incomplete Dataset

Consider a set of Bellman equations:

$$\begin{bmatrix} Q(s_0, a_0) = r_1 + \gamma Q(s_1, a_1') \\ Q(s_1, a_1) = r_2 + \gamma Q(s_2, a_2') \\ \vdots \\ Q(s_{N-1}, a_{N-1}) = r_N + \gamma Q(s_N, a_N') \end{bmatrix}$$

Bellman Equations over an Incomplete Dataset

Consider a set of Bellman equations:

$$\begin{bmatrix} Q(s_0, a_0) = r_1 + \gamma Q(s_1, a_1') \\ Q(s_1, a_1) = r_2 + \gamma Q(s_2, a_2') \\ \vdots \\ Q(s_{N-1}, a_{N-1}) = r_N + \gamma Q(s_N, a_N') \end{bmatrix}$$

N Variables

Bellman Equations over an Incomplete Dataset

Consider a set of Bellman equations:

$$\begin{bmatrix} Q(s_0, a_0) = r_1 + \gamma Q(s_1, a_1') \\ Q(s_1, a_1) = r_2 + \gamma Q(s_2, a_2') \\ \vdots \\ Q(s_{N-1}, a_{N-1}) = r_N + \gamma Q(s_N, a_N') \end{bmatrix}$$

N Variables

N Variables

Bellman Equations over an Incomplete Dataset

Consider a set of Bellman equations:

$$\left[\begin{array}{r} Q(s_0, a_0) = r_1 + \gamma Q(s_1, a_1') \\ Q(s_1, a_1) = r_2 + \gamma Q(s_2, a_2') \\ \vdots \\ Q(s_{N-1}, a_{N-1}) = r_N + \gamma Q(s_N, a_N') \end{array}\right] \text{N Equations}$$

N Variables + N Variables = 2N Variables

An underdetermined system = infinite solutions

Bellman Equations over an Incomplete Dataset

Consider a set of Bellman equations:

$$\left[\begin{array}{r} Q(s_0, a_0) = r_1 + \gamma Q(s_1, a_1') \\ Q(s_1, a_1) = r_2 + \gamma Q(s_2, a_2') \\ \vdots \\ Q(s_{N-1}, a_{N-1}) = r_N + \gamma Q(s_N, a_N') \end{array}\right] \text{N Equations}$$

N Variables + N Variables = 2N Variables

An underdetermined system = infinite solutions

Problem: Not every solution has low value error.

Bellman Equations over an Incomplete Dataset

Consider a single transition (where $\boldsymbol{r} = \mathbf{0}$):

$$Q(s, a) = 0 + \gamma Q(s', a')$$

If we set $Q(s', a') = 100$ then $Q(s, a) = 100\gamma$

If we set $Q(s', a') = 0$ then $Q(s, a) = 0$

In both cases the Bellman error is 0, but the value prediction is very different.

In our experiments we show…

When working with off-policy datasets…

Minimizing the Bellman error finds value functions with low Bellman error but high value error.

$\Rightarrow$ The Bellman error is not a meaningful off-policy objective.

Summary

(1) The magnitude of the Bellman error is influenced heavily by bias.
(2) Low Bellman error has little significance if there is missing data.

⇒ The Bellman error is not a good proxy for value error.

Additional insights, analysis, and experiments in the paper.

Thanks for listening!