The O.D.E. method for Convergence of Stochastic Approximation and Reinforcement Learning

Ravi Kiran Raman       Suryanarayana Sankagiri

University of Illinois
Urbana Champaign

March 07, 2018
The authors

Figure: Prof. Vivek Borkar
IIT Bombay, India

Figure: Prof. Sean Meyn
University of Florida

Why study this paper?
Outline of the paper

- What is Stochastic Approximation?

  It is the algorithm given by the recursion

  \[ X(n+1) = X(n) + a(n) \cdot [h(X(n)) + M(n+1)], \]

  \( n \geq 0 \)

- When does Stochastic Approximation converge?

  The algorithm 'tracks' the ODE

  \[ \dot{x} = h(x(t)) \]

  Perturbation due to noise and discretization decreases with decaying

  step size

  ODE converges \( \Rightarrow \) algorithm converges!

- Why Reinforcement Learning algorithms can be treated as

  Stochastic Approximation?

  That's what we see next!
Outline of the paper

- What is Stochastic Approximation?
  - It is the algorithm given by the recursion
    \[ X(n + 1) = X(n) + a(n)[h(X(n)) + M(n + 1)], \quad n \geq 0 \]
  - \( X(n) \in \mathbb{R}^d \), \( h: \mathbb{R}^d \to \mathbb{R}^d \), \( M(n) \) is additive noise

- When does Stochastic Approximation converge?
  - The algorithm 'tracks' the ODE \( \dot{x} = h(x(t)) \)
  - Perturbation due to noise and discretization decreases with decaying step size
  - ODE converges \( \Rightarrow \) algorithm converges!

- Why Reinforcement Learning algorithms can be treated as Stochastic Approximation?
  - That's what we see next!
Outline of the paper

- What is Stochastic Approximation?
  - It is the algorithm given by the recursion
    \[ X(n + 1) = X(n) + a(n)[h(X(n)) + M(n + 1)], \quad n \geq 0 \]
  - \( X(n) \in \mathbb{R}^d, h : \mathbb{R}^d \to \mathbb{R}^d, M(n) \) is additive noise

- When does Stochastic Approximation converge?
  - The algorithm ‘tracks’ the ODE \( \dot{x} = h(x(t)) \)
  - Perturbation due to noise and discretization decreases with decaying step size
  - ODE converges \( \Rightarrow \) algorithm converges!

- Why Reinforcement Learning algorithms can be treated as Stochastic Approximation?
  - That’s what we see next!
Outline of the paper

- What is Stochastic Approximation?
  - It is the algorithm given by the recursion
    \[ X(n + 1) = X(n) + a(n)[h(X(n)) + M(n + 1)], \quad n \geq 0 \]
  - \( X(n) \in \mathbb{R}^d, h : \mathbb{R}^d \to \mathbb{R}^d, M(n) \) is additive noise

- When does Stochastic Approximation converge?
  - The algorithm ‘tracks’ the ODE \( \dot{x} = h(x(t)) \)
  - Perturbation due to noise and discretization decreases with decaying step size
  - ODE converges \( \Rightarrow \) algorithm converges!

- Why Reinforcement Learning algorithms can be treated as Stochastic Approximation?
  - That’s what we see next!
Outline of the paper

• What is Stochastic Approximation?
  ○ It is the algorithm given by the recursion

\[ X(n + 1) = X(n) + a(n)[h(X(n)) + M(n + 1)], \quad n \geq 0 \]

  ○ \( X(n) \in \mathbb{R}^d, h : \mathbb{R}^d \to \mathbb{R}^d, M(n) \) is additive noise

• When does Stochastic Approximation converge?
  ○ The algorithm ‘tracks’ the ODE \( \dot{x} = h(x(t)) \)
Outline of the paper

• What is Stochastic Approximation?
  ○ It is the algorithm given by the recursion

\[ X(n + 1) = X(n) + a(n)[h(X(n)) + M(n + 1)], \quad n \geq 0 \]

  ○ \( X(n) \in \mathbb{R}^d, \ h : \mathbb{R}^d \rightarrow \mathbb{R}^d, \) \( M(n) \) is additive noise

• When does Stochastic Approximation converge?
  ○ The algorithm ‘tracks’ the ODE \( \dot{x} = h(x(t)) \)
  ○ Perturbation due to noise and discretization decreases with decaying step size
Outline of the paper

• What is Stochastic Approximation?
  ○ It is the algorithm given by the recursion

\[ X(n + 1) = X(n) + a(n)[h(X(n)) + M(n + 1)], \quad n \geq 0 \]

  ○ \(X(n) \in \mathbb{R}^d, h : \mathbb{R}^d \rightarrow \mathbb{R}^d, M(n)\) is additive noise

• When does Stochastic Approximation converge?
  ○ The algorithm ‘tracks’ the ODE \(\dot{x} = h(x(t))\)
  ○ Perturbation due to noise and discretization decreases with decaying step size
  ○ ODE converges \(\Rightarrow\) algorithm converges!
Outline of the paper

• What is Stochastic Approximation?
  ○ It is the algorithm given by the recursion

\[ X(n + 1) = X(n) + a(n)[h(X(n)) + M(n + 1)], \quad n \geq 0 \]

  ○ \( X(n) \in \mathbb{R}^d, h : \mathbb{R}^d \to \mathbb{R}^d, M(n) \) is additive noise

• When does Stochastic Approximation converge?
  ○ The algorithm ‘tracks’ the ODE \( \dot{x} = h(x(t)) \)
  ○ Perturbation due to noise and discretization decreases with decaying step size
  ○ ODE converges \( \Rightarrow \) algorithm converges!

• Why Reinforcement Learning algorithms can be treated as Stochastic Approximation?
Outline of the paper

• What is Stochastic Approximation?
  ○ It is the algorithm given by the recursion

\[ X(n + 1) = X(n) + a(n)[h(X(n)) + M(n + 1)], \ n \geq 0 \]

  ○ \( X(n) \in \mathbb{R}^d, h : \mathbb{R}^d \rightarrow \mathbb{R}^d, M(n) \) is additive noise

• When does Stochastic Approximation converge?
  ○ The algorithm ‘tracks’ the ODE \( \dot{x} = h(x(t)) \)
  ○ Perturbation due to noise and discretization decreases with decaying step size
  ○ ODE converges \( \Rightarrow \) algorithm converges!

• Why Reinforcement Learning algorithms can be treated as Stochastic Approximation?
  ○ That’s what we see next!
MDP & Reinforcement Learning

• Learn optimal control policy (actions at a state) to minimize loss, based on the costs incurred in taking the action (Chess, Go etc.)
MDP & Reinforcement Learning

• Learn optimal control policy (actions at a state) to minimize loss, based on the costs incurred in taking the action (Chess, Go etc.)

• Markov Decision Process
  ○ State space $S = \{1, \ldots, |S|\}$
  ○ Action space $U = \{1, \ldots, |U|\}$
  ○ Control Markov chain: state transition kernel, $P_{i,j}(u)$
  ○ Cost of action $u$ at state $i$ (independent random variable) $C(i, u)$
  ○ Discount factor $\alpha \in (0, 1)$
MDP & Reinforcement Learning

- Learn optimal control policy (actions at a state) to minimize loss, based on the costs incurred in taking the action (Chess, Go etc.)

- Markov Decision Process
  - State space $S = \{1, \ldots, |S|\}$
  - Action space $U = \{1, \ldots, |U|\}$
  - Control Markov chain: state transition kernel, $P_{i,j}(u)$
  - Cost of action $u$ at state $i$ (independent random variable) $C(i, u)$
  - Discount factor $\alpha \in (0, 1)$

- Discounted cost: $\lim_{T \to \infty} \sum_{t=1}^{T} \alpha^t C(X_t, U_t)$

- Learning problem:

$$J^*(i) = \inf_{\mu_1, \mu_2, \ldots} \lim_{T \to \infty} \mathbb{E} \left[ \sum_{t=1}^{T} \alpha^t C(X_t, U_t) | X_0 = i \right] \quad (1)$$
Value Iteration & Convergence

- Value function of MDP (dynamic programming formulation),

\[ J^*(i) = \min_u \mathbb{E}[C(i, u) + \alpha \sum_{j \in S} P_{i,j}(u)J^*(j)|X_0 = i] \]  \hspace{1cm} (2)

- Equivalently, (2) can be written as \( J^* = T(J^*) \)
Value Iteration & Convergence

- Value function of MDP (dynamic programming formulation),

\[
J^*(i) = \min_u \mathbb{E}[C(i, u) + \alpha \sum_{j \in S} P_{i,j}(u) J^*(j)|X_0 = i]
\]  

(2)

- Equivalently, (2) can be written as \( J^* = T(J^*) \)

- Estimate using value iteration \( \hat{J}_{t+1} = T(\hat{J}_t) \)

- Optimal policy: actions minimizing value function at each state
Value Iteration & Convergence

- Value function of MDP (dynamic programming formulation),
  \[ J^*(i) = \min_u \mathbb{E}[C(i, u) + \alpha \sum_{j \in S} P_{i,j}(u) J^*(j) | X_0 = i] \]  
  (2)

- Equivalently, (2) can be written as \( J^* = T(J^*) \)

- Estimate using value iteration \( \hat{J}_{t+1} = T(\hat{J}_t) \)

- Optimal policy: actions minimizing value function at each state

- \( T \) is a contraction under \( \| \cdot \|_\infty \) with parameter \( \alpha \)

- Linear convergence rate by Banach’s fixed point theorem
  \[ \| \hat{J}_t - J^* \|_\infty \leq \frac{\alpha^t}{1 - \alpha} \| \hat{J}_1 - \hat{J}_0 \|_\infty \]  
  (3)
Q-function

In practice,

- minimization at each iteration is computationally expensive
- transition kernel structure not fully known
- state-action space is too large to store, recover, and compute
Q-function

In practice,

• minimization at each iteration is computationally expensive
• transition kernel structure not fully known
• state-action space is too large to store, recover, and compute

Q-function: a state-action cost matrix

\[ Q(i, u) = C(i, u) + \alpha \sum_{j \in S} P_{i,j}(u) \min_{\tilde{u}} Q(j, \tilde{u}) \]  \hspace{1cm} (4)
Q-learning

- Value iteration:

\[ \hat{Q}_{t+1}(i, u) = \bar{C}(i, u) + \alpha \sum_{j \in S} P_{i,j}(u) \min_{\tilde{u}} \hat{Q}_t(j, \tilde{u}), \quad \forall (i, u) \]
Q-learning

• Value iteration:

\[ \hat{Q}_{t+1}(i, u) = \bar{C}(i, u) + \alpha \sum_{j \in S} P_{i,j}(u) \min_{\tilde{u}} \hat{Q}_t(j, \tilde{u}), \ \forall (i, u) \]

• Synchronous: if \( X_t = i, U_t = u, X_{t+1} = j \) for each state,

\[ \hat{Q}_{t+1}(i, u) = (1 - \epsilon_t) \hat{Q}_t(i, u) + \epsilon_t (\bar{C}(i, u) + \alpha \min_{\tilde{u}} \hat{Q}_t(j, \tilde{u})), \ \forall (i, u) \]
Q-learning

- Value iteration:
  \[ \hat{Q}_{t+1}(i, u) = \bar{C}(i, u) + \alpha \sum_{j \in S} P_{i,j}(u) \min_{\tilde{u}} \hat{Q}_t(j, \tilde{u}), \ \forall (i, u) \]

- Synchronous: if \( X_t = i, U_t = u, X_{t+1} = j \) for each state,
  \[ \hat{Q}_{t+1}(i, u) = (1 - \epsilon_t) \hat{Q}_t(i, u) + \epsilon_t (C(i, u) + \alpha \min_{\tilde{u}} \hat{Q}_t(j, \tilde{u})), \ \forall (i, u) \]

- Asynchronous: \( X_t = i, U_t = u, X_{t+1} = j \)
  \[ \hat{Q}_{t+1}(i, u) = (1 - \epsilon_t) \hat{Q}_t(i, u) + \epsilon_t (C(i, u) + \alpha \min_{\tilde{u}} \hat{Q}_t(j, \tilde{u})), \]
  \[ \hat{Q}_{t+1}(k, u) = \hat{Q}_t(k, v), \ \forall (k, v) \neq (i, u) \]
Stochastic Approximation

Stochastic approximation problem:

\[ X(n + 1) = X(n) + a(n)[h(X(n)) + M(n + 1)], \quad n \geq 0 \]  \hspace{1cm} (5)

where \( X(n) \in \mathbb{R}^d \),

\[ h : \mathbb{R}^d \rightarrow \mathbb{R}^d, \]

\[ a(n) > 0, \quad n \geq 0, \]

\( M(n) \) zero mean Martingale Difference Sequence
Stochastic Approximation

Stochastic approximation problem:

\[ X(n + 1) = X(n) + a(n)[h(X(n)) + M(n + 1)], \quad n \geq 0 \tag{5} \]

where \( X(n) \in \mathbb{R}^d \),

\[ h : \mathbb{R}^d \to \mathbb{R}^d, \]

\[ a(n) > 0, \quad n \geq 0, \]

\( M(n) \) zero mean Martingale Difference Sequence

Under some conditions, iterates track solution of the associated ODE:

\[ \dot{x}(t) = h(x(t)) \tag{6} \]
Q-learning to Stochastic Approximation

Q-learning can be formulated as a stochastic approximation problem as follows:

\[ X(n) = \hat{Q}_n, \quad a(n) = \epsilon_n, \]

\[ h(Q)_{i,u} = \bar{C}(i, u) - Q(i, u) + \alpha \sum_{j \in S} P_{i,j}(u) \min_{v} Q(j, v) \]

\[ M(n)_{i,u} = C(i, u) - \bar{C}(i, u) + \alpha \left[ \min_{v} Q(j, v) - \sum_{j \in S} P_{i,j}(u) \min_{v} Q(j, v) \right] \]
Q-learning to Stochastic Approximation

Q-learning can be formulated as a stochastic approximation problem as follows:

\[ X(n) = \hat{Q}_n, \quad a(n) = \epsilon_n, \]

\[ h(Q)_{i,u} = \bar{C}(i, u) - Q(i, u) + \alpha \sum_{j \in S} P_{i,j}(u) \min_{\nu} Q(j, \nu) \]

\[ M(n)_{i,u} = C(i, u) - \bar{C}(i, u) + \alpha \left[ \min_{\nu} Q(j, \nu) - \sum_{j \in S} P_{i,j}(u) \min_{\nu} Q(j, \nu) \right] \]

Then, Q-function can be defined by the point the solution to the following ODE converges to:

\[ \dot{Q} = h(Q) \quad (7) \]
Stochastic Gradient Descent (a side note)

Typical SGD iteration update can be modeled as:

\[
x_{n+1} = x_n - \epsilon_n \hat{\nabla} f(x_n)
= x_n + \epsilon_n \left[ -\nabla f(x_n) + \left( \nabla f(x_n) - \hat{\nabla} f(x_n) \right) \right]
\]

Thus, under some assumptions on the function (derivatives), iterates track solution to

\[
\dot{x}(t) = -\nabla f(x(t))
\]
Convergence of Stochastic Approximation

Recall

The central theme in the ODE method

The stochastic approximation algorithm can be approximated by solutions to the underlying ODE

Therefore, it is reasonable to expect a result as follows:

Theorem (loose statement)

Suppose the ODE $\dot{x} = h(x(t))$ has a unique globally asymptotically stable equilibrium $x^*$. Then $X(n)$ → $x^*$ with probability one for any initial condition $X(0) \in \mathbb{R}^d$

Theorem is useful if it is easy to show asymptotic stability of the underlying ODE
Convergence of Stochastic Approximation

Recall
The central theme in the ODE method
Convergence of Stochastic Approximation

Recall
The central theme in the ODE method
The stochastic approximation algorithm can be approximated by
solutions to the underlying ODE
Convergence of Stochastic Approximation

Recall
The central theme in the ODE method
The stochastic approximation algorithm can be approximated by solutions to the underlying ODE

Therefore, it is reasonable to expect a result as follows:
Convergence of Stochastic Approximation

Recall

The central theme in the ODE method
The stochastic approximation algorithm can be approximated by solutions to the underlying ODE

Therefore, it is reasonable to expect a result as follows:

Theorem (loose statement)
Suppose the ODE \( \dot{x} = h(x(t)) \) has a unique globally asymptotically stable equilibrium \( x^* \). Then \( X(n) \rightarrow x^* \) with probability one for any initial condition \( X(0) \in \mathbb{R}^d \).
Convergence of Stochastic Approximation

Recall
The central theme in the ODE method
The stochastic approximation algorithm can be approximated by solutions to the underlying ODE

Therefore, it is reasonable to expect a result as follows:

**Theorem (loose statement)**

Suppose the ODE $\dot{x} = h(x(t))$ has a unique globally asymptotically stable equilibrium $x^*$. Then $X(n) \to x^*$ with probability one for any initial condition $X(0) \in \mathbb{R}^d$

Theorem is useful if it is easy to show asymptotic stability of the underlying ODE
Proof sketch

No theorem comes without assumptions and lemmas!
Proof sketch

No theorem comes without assumptions and lemmas!

Proof sketch $\Rightarrow$ Technical Assumptions $\Rightarrow$ Main contribution
Proof sketch

No theorem comes without assumptions and lemmas!

Proof sketch $\Rightarrow$ Technical Assumptions $\Rightarrow$ Main contribution

The main contribution of the paper is to give an alternate, simpler proof for establishing a key lemma for the above theorem
Proof sketch

No theorem comes without assumptions and lemmas!

Proof sketch $\Rightarrow$ Technical Assumptions $\Rightarrow$ Main contribution

The main contribution of the paper is to give an alternate, simpler proof for establishing a key lemma for the above theorem

Are you ready for the proof?
The picture is taken from the cover of Vivek S Borkar. “Stochastic approximation: A Dynamical Systems Viewpoint”. In: Cambridge Books (2008)
Some constructions

- Divide time into blocks of length approximately $T$, for some $T > 0$
Some constructions

- Divide time into blocks of length approximately \( T \), for some \( T > 0 \)
  - \( t(0) = T(0) = 0; t(n) = \sum_{i<n} a(i) \)
Some constructions

- Divide time into blocks of length approximately $T$, for some $T > 0$
  - $t(0) = T(0) = 0$; $t(n) = \sum_{i < n} a(i)$
  - $T(n + 1) = \min\{t(j) : t(j) > T(n) + T\}$
Some constructions

• Divide time into blocks of length approximately \( T \), for some \( T > 0 \)
  
  \[
  \begin{align*}
  t(0) &= T(0) = 0; \\
  t(n) &= \sum_{i<n} a(i) \\
  T(n+1) &= \min\{t(j) : t(j) > T(n) + T\} \
  \end{align*}
  \]

  ○ To do this recursively, we need \( \sum_n a(n) = \infty \)
Some constructions

- Divide time into blocks of length approximately $T$, for some $T > 0$
  - $t(0) = T(0) = 0; t(n) = \sum_{i<n} a(i)$
  - $T(n+1) = \min\{t(j) : t(j) > T(n) + T\}$
  - To do this recursively, we need $\sum_n a(n) = \infty$

- Now construct two continuous time functions, which ultimately come close together
Some constructions

- Divide time into blocks of length approximately $T$, for some $T > 0$
  - $t(0) = T(0) = 0; t(n) = \sum_{i<n} a(i)$
  - $T(n + 1) = \min\{t(j) : t(j) > T(n) + T\}$
  - To do this recursively, we need $\sum_n a(n) = \infty$

- Now construct two continuous time functions, which ultimately come close together
  - $\{\psi(t), t > 0\}$ is defined s.t. $\psi(t(n)) = X(n)$ with linear interpolation on $[t(n), t(n + 1)]$ (continuous function)
Some constructions

- Divide time into blocks of length approximately $T$, for some $T > 0$
  - $t(0) = T(0) = 0; t(n) = \sum_{i<n} a(i)$
  - $T(n + 1) = \min\{t(j) : t(j) > T(n) + T\}$
  - To do this recursively, we need $\sum_n a(n) = \infty$

- Now construct two continuous time functions, which ultimately come close together
  - $\{\psi(t), t > 0\}$ is defined s.t. $\psi(t(n)) = X(n)$ with linear interpolation on $[t(n), t(n + 1)]$ (continuous function)
  - $\{\hat{\psi}(t), t > 0\}$ is defined s.t. $\hat{\psi}(T(n)) = \psi(T(n))$ and $\hat{\psi}(t)$ evolves according to the ODE $\dot{x} = h(x)$ on the interval $[T(j), T(j + 1))$ (piece-wise continuous function)
Proof Sketch

• What can we infer from the globally asymptotic stability property of $h$?
Proof Sketch

• What can we infer from the globally asymptotic stability property of $h$?
Proof Sketch

• What can we infer from the globally asymptotic stability property of $h$? Basic control theory tells us:
  ◦ solution of the ODE reaches close to $x^*$ with time
Proof Sketch

- What can we infer from the globally asymptotic stability property of $h$? Basic control theory tells us:
  - solution of the ODE reaches close to $x^*$ with time
  - If $X(n)$ lies in a bounded set $B \forall n$, we can choose $T$ large enough s.t. $\hat{\psi}(T(n))$ lies close to the equilibrium $\forall n$
Proof Sketch

• What can we infer from the globally asymptotic stability property of \( h \)? Basic control theory tells us:
  ○ solution of the ODE reaches close to \( x^* \) with time
  ○ If \( X(n) \) lies in a bounded set \( B \ \forall \ n \), we can choose \( T \) large enough s.t. \( \hat{\psi}(T(j)^-) \) lies close to the equilibrium \( \forall \ j \)
  ○ If \( \psi(T(n)) = \hat{\psi}(T(n)) \) is within a small ball of radius \( \delta \) around \( x^* \), \( \hat{\psi}(t) \) always remains within a small ball of radius \( \epsilon \)
Proof Sketch

- We would like to establish:

\[
\sup_{t \geq T} \| \psi(t) - \hat{\psi}(t) \| \xrightarrow{T \to \infty} 0
\]

which allows us to make the following deductions in the asymptotic limit:

- \( \hat{\psi}(T(j)) \) lies close to \( x^* \)
- \( \psi(T(j)) \) close to \( x^* \)
- Stability \( \Rightarrow \hat{\psi}(t) \) close to the equilibrium \( \forall t \in [T(j), T(j+1)) \)
- The two functions are close \( \Rightarrow \psi(t) \) close to \( x^* \) \( \forall t \in [T(j), T(j+1)) \)

This implies \( X(n) \) remains close to \( x^* \) in the asymptotic limit
Proof Sketch

• We would like to establish:

\[
\sup_{t \geq T} \| \psi(t) - \hat{\psi}(t) \| \xrightarrow{T \to \infty} 0
\]

which allows us to make the following deductions in the asymptotic limit:

○ \( \hat{\psi}(T(j)^-) \) lies close to \( x^* \) \( \Rightarrow \) \( \psi(T(j)^-) \) close to \( x^* \)
Proof Sketch

• We would like to establish:

\[ \sup_{t \geq T} \| \psi(t) - \hat{\psi}(t) \| \xrightarrow{T \to \infty} 0 \]

which allows us to make the following deductions in the asymptotic limit:

○ \( \hat{\psi}(T(j)^-) \) lies close to \( x^* \) \( \Rightarrow \) \( \psi(T(j)^-) \) close to \( x^* \)

○ \( \psi \) continuous \( \Rightarrow \) \( \psi(T(j)) = \hat{\psi}(T(j)) \) close to \( x^* \)
Proof Sketch

- We would like to establish:

\[
\sup_{t \geq T} \| \psi(t) - \hat{\psi}(t) \| \xrightarrow{T \to \infty} 0
\]

which allows us to make the following deductions in the asymptotic limit:

- \( \hat{\psi}(T(j)^-) \) lies close to \( x^* \) \( \Rightarrow \) \( \psi(T(j)^-) \) close to \( x^* \)
- \( \psi \) continuous \( \Rightarrow \psi(T(j)) = \hat{\psi}(T(j)) \) close to \( x^* \)
- Stability \( \Rightarrow \hat{\psi}(t) \) close to the equilibrium \( \forall \; t \in [T(j), T(j + 1)) \)
Proof Sketch

- We would like to establish:

\[
\sup_{t \geq T} \| \psi(t) - \hat{\psi}(t) \| \xrightarrow{T \to \infty} 0
\]

which allows us to make the following deductions in the asymptotic limit:

- \( \hat{\psi}(T(j)^-) \) lies close to \( x^* \) \( \Rightarrow \) \( \psi(T(j)^-) \) close to \( x^* \)
- \( \psi \) continuous \( \Rightarrow \) \( \psi(T(j)) = \hat{\psi}(T(j)) \) close to \( x^* \)
- Stability \( \Rightarrow \) \( \hat{\psi}(t) \) close to the equilibrium \( \forall \ t \in [T(j), T(j + 1)) \)
- The two functions are close \( \Rightarrow \) \( \psi(t) \) close to \( x^* \) \( \forall \ t \in [T(j), T(j + 1)) \)
Proof Sketch

• We would like to establish:

$$\sup_{t \geq T} \|\psi(t) - \hat{\psi}(t)\| \xrightarrow{T \to \infty} 0$$

which allows us to make the following deductions in the asymptotic limit:

○ $\hat{\psi}(T(j)^-) \text{ lies close to } x^* \Rightarrow \psi(T(j)^-) \text{ close to } x^*$
○ $\psi$ continuous $\Rightarrow \psi(T(j)) = \hat{\psi}(T(j)) \text{ close to } x^*$
○ Stability $\Rightarrow \hat{\psi}(t) \text{ close to the equilibrium } \forall \ t \in [T(j), T(j + 1))$
○ The two functions are close $\Rightarrow \psi(t) \text{ close to } x^* \ \forall \ t \in [T(j), T(j + 1))$
○ This implies $X(n)$ remains close to $x^*$ in the asymptotic limit
Some standard assumptions

To establish \( \sup_{t \geq T} \| \psi(t) - \hat{\psi}(t) \| \xrightarrow{T \to \infty} 0 \), we need some additional assumptions:

- **Tapering stepsizes** \( \{a(n)\} \) should satisfy \( \sum_n a(n)^2 < \infty \)
- **Bounded noise** \( \{M(n)\} \) should satisfy:

  \[
  \mathbb{E}[\| M_{n+1} \|^2 | M_1, \ldots, M_n] \leq C(1 + \| X(n) \|^2) \]
  
  for some finite \( C \)

- Coupled with the previous assumption that \( \sup_n \| X(n) \| < \infty \), we can show that the error due to noise goes to zero:

  \[
  \mathbb{E}[\| M_{n+1} \|^2 | M_1, \ldots, M_n] \leq C' \]
  
  for some finite \( C' \)

  \[
  \lim_{m \to \infty} \sum_{n \geq m} a(n)^2 \mathbb{E}[\| M_{n+1} \|^2 | M_1, \ldots, M_n] = 0
  \]

- **Lipschitz** the function \( h \) must be Lipschitz, to drive the discretization error to zero
Convergence of Stochastic Approximation

Theorem 2.2 (Convergence of SA)
Suppose the previous assumptions hold.

Suppose further that the ODE \( \dot{x} = h(x(t)) \) has a unique globally asymptotically stable equilibrium \( x^* \).

Then \( X(n) \to x^* \) with probability one for any initial condition \( X(0) \in \mathbb{R}^d \).

We skip a formal proof of the above theorem, referring to the main paper and Chapter 2 of the book.

---

Convergence of Stochastic Approximation

Theorem 2.2 (Convergence of SA)
Suppose the previous assumptions hold. Suppose further that the ODE $\dot{x} = h(x(t))$ has a unique globally asymptotically stable equilibrium $x^*$. 

---

Convergence of Stochastic Approximation

Theorem 2.2 (Convergence of SA)

Suppose the previous assumptions hold. Suppose further that the ODE $\dot{x} = h(x(t))$ has a unique globally asymptotically stable equilibrium $x^\ast$. Then $X(n) \to x^\ast$ with probability one for any initial condition $X(0) \in \mathbb{R}^d$

We skip a formal proof of the above theorem, referring to the main paper and Chapter 2 of the book\textsuperscript{2}

---

Asynchronous Stochastic Approximation

- Stochastic Approximation can be asynchronous in practical applications! E.g. asynchronous Q-learning
- Asynchronous updates means each component of $X(n)$ is updated by a separate processor
- Each processor has different speed:
  - Let $\nu(i, n)$ be the number of updates executed by the processor up to time $n$
  - We assume $\nu(i, n)/n \geq \Delta$ for large $n$, where $\Delta > 0$ is deterministic
- Further, there could be interprocessor communication delays between processors $\tau_{kj}(n), 1 \leq k, j, \leq d$
  - Processor ($k$) may use the data for $X_j(m)$ only for $m \leq n - \tau_{kj}(n)$
  - We assume $\tau_{kj}(n) \leq \bar{\tau} < \infty$
Asynchronous Stochastic Approximation

- The result in the previous paper showed that asynchronous stochastic approximation converges if the iterates are assumed to be bounded!
- The result in this paper completes the above, by demonstrating the iterates are bounded

Theorem 2.5 (Convergence on Asynchronous Stochastic Approximation)

Under the conditions of listed above, the asynchronous iterates remain bonded with probability 1, and therefore converge to $x^*$ with probability 1
Bounded Iterates

Theorem 2.1 (Stability)
Under tapering stepsizes, bounded noise, Lipschitz continuity of $h$, and existence of $h_\infty$, if the origin is an asymptotically stable equilibrium of the limiting ODE,

$$\sup_n \|X(n)\| < \infty, \text{ almost surely}$$

(8)
Scaled ODEs

- Original SA ODE:
  \[ \dot{x}(t) = h(x(t)) \]  
  (9)
Scaled ODEs

- Original SA ODE:
  \[ \dot{x}(t) = h(x(t)) \quad (9) \]

- Scaled ODE:
  \[ \dot{x}(t) = h_r(x(t)) = \frac{h(rx(t))}{r} \quad (10) \]
Scaled ODEs

• Original SA ODE:
\[ \dot{x}(t) = h(x(t)) \]  \hspace{1cm} (9)

• Scaled ODE:
\[ \dot{x}(t) = h_r(x(t)) = \frac{h(rx(t))}{r} \]  \hspace{1cm} (10)

• Limiting ODE:
\[ \dot{x}(t) = h_\infty(x(t)) = \lim_{r \to \infty} h_r(x(t)) \]  \hspace{1cm} (11)
Very Rough Proof Sketch

- Limiting ODE is globally exponentially asymptotically stable: follows from \( h_\infty(cx) = ch_\infty(x), c > 0 \)
Very Rough Proof Sketch

- Limiting ODE is globally exponentially asymptotically stable: follows from $h_\infty(cx) = ch_\infty(x), c > 0$
- For $T > 0$ large enough, solution is in an $\epsilon/2$ ball around origin
Very Rough Proof Sketch

- Limiting ODE is globally exponentially asymptotically stable: follows from $h_\infty(cx) = ch_\infty(x)$, $c > 0$
- For $T > 0$ large enough, solution is in an $\epsilon/2$ ball around origin
- Lipschitz $h$: solns $\hat{\phi}(t)$, $\phi_\infty(t)$ of (10), (11) are bounded
Very Rough Proof Sketch

- Limiting ODE is globally exponentially asymptotically stable: follows from $h_\infty(cx) = ch_\infty(x), c > 0$
- For $T > 0$ large enough, solution is in an $\epsilon/2$ ball around origin
- Lipschitz $h$: solns $\hat{\phi}(t), \phi_\infty(t)$ of (10), (11) are bounded
- $h_r \rightarrow h_\infty$ uniformly: for $r > R, t > T$, $\|\hat{\phi}(t) - \phi_\infty(t)\| \leq \epsilon/2$
Very Rough Proof Sketch

- Limiting ODE is globally exponentially asymptotically stable: follows from \( h_\infty(cx) = ch_\infty(x), c > 0 \)
- For \( T > 0 \) large enough, solution is in an \( \epsilon/2 \) ball around origin
- Lipschitz \( h \): solns \( \hat{\phi}(t), \phi_\infty(t) \) of (10), (11) are bounded
- \( h_r \to h_\infty \) uniformly: for \( r > R, t > T \), \( \|\hat{\phi}(t) - \phi_\infty(t)\| \leq \epsilon/2 \)
- For \( R, T \) large enough, solution to scaled ODE is within an \( \epsilon \)-ball of origin
• $\phi(t)$: linearly interpolate iterates piecewise over $[T(j), T(j + 1))$ with $\phi(T(j)) = \frac{X(j)}{\max(1, X(m(j)))}$
Very Rough Proof Sketch

- \( \phi(t) \): linearly interpolate iterates piecewise over \([T(j), T(j + 1))\) with \( \phi(T(j)) = \frac{X(j)}{\max(1, X(m(j)))} \)

- Let \( \hat{\phi}(t) \) be piecewise continuous solutions of (10) on \([T(j), T(j + 1))\) with \( \hat{\phi}(T(j)) = \phi(T(j)) \)
Very Rough Proof Sketch

- $\phi(t)$: linearly interpolate iterates piecewise over $[T(j), T(j + 1))$ with $\phi(T(j)) = \frac{X(j)}{\max(1, X(m(j)))}$
- Let $\hat{\phi}(t)$ be piecewise continuous solutions of (10) on $[T(j), T(j + 1))$ with $\hat{\phi}(T(j)) = \phi(T(j))$
- Then $\|\phi(t) - \hat{\phi}(t)\| \to 0$ as $t \to \infty$, $\implies \sup_t \|\phi(t)\| < \infty$
Very Rough Proof Sketch

- \( \phi(t) \): linearly interpolate iterates piecewise over \([T(j), T(j + 1)]\) with \( \phi(T(j)) = \frac{X(j)}{\max(1, X(m(j)))} \)

- Let \( \hat{\phi}(t) \) be piecewise continuous solutions of (10) on \([T(j), T(j + 1)]\) with \( \hat{\phi}(T(j)) = \phi(T(j)) \)

- Then \( \|\phi(t) - \hat{\phi}(t)\| \to 0 \) as \( t \to \infty \), \( \implies \sup_t \|\phi(t)\| < \infty \)

- Thus, the iterates of SA are bounded
Back to Q-learning

We know that in the SA formulation,

\[ h(Q)_{i,u} = \bar{C}(i, u) - Q(i, u) + \alpha \sum_{j \in S} P_{i,j}(u) \min_{v} Q(j, v) \]  \hspace{1cm} (12)

Then, the limiting ODE is characterized by

\[ h_{\infty}(Q)_{i,u} = \alpha \sum_{j \in S} P_{i,j}(u) \min_{v} Q(j, v) - Q(i, u) = F_{\infty}(Q) - Q \]  \hspace{1cm} (13)

- \( F_{\infty}(\cdot) \) is a contraction mapping under \( \| \cdot \|_{\infty} \)
- Corresponding ODE has unique equilibrium point and is asymptotically stable
- Proof of convergence follows
Takeback

• Prove stability of iterates of stochastic approximation (synchronous and asynchronous)
• Proof of convergence of asynchronous stochastic approximation without prior assumptions on stability of iterates
• Proof of convergence of Q-learning through theory of SA
Takeback

• Prove stability of iterates of stochastic approximation (synchronous and asynchronous)
• Proof of convergence of asynchronous stochastic approximation without prior assumptions on stability of iterates
• Proof of convergence of Q-learning through theory of SA

Some aspects not discussed here:
• Stochastic approximation under constant, bounded stepsize.
• Convergence rate of SA
• Adaptive critic-type algorithms