Multicoloured Hardcore Model: Fast Mixing and Queueing

Sam Olesker-Taylor
University of Warwick

June 2024

1. Set-Up, Motivation and Examples
2. Model and Main Result
3. Application as a Queueing System
4. Simulations

Set-Up, Motivation and Examples

Set-Up, Motivation and Examples

Set-Up, Motivation and Examples

Set-Up. Let $G = (V, E)$ be a finite graph

- Our objective is to *properly colour* a (large) subset U of vertices
- If $u, u' \in U$ and $\{u, u'\} \in E$, then u and u' have different colours
- one colour $\leftrightsquigarrow$ independent sets, $U = V \leftrightsquigarrow$ proper colourings

Set-Up, Motivation and Examples

Set-Up. Let $G = (V, E)$ be a finite graph

- Our objective is to *properly colour* a (large) subset U of vertices
- If $u, u' \in U$ and $\{u, u'\} \in E$, then u and u' have different colours
- one colour $\leftrightsquigarrow$ independent sets, $U = V \leftrightsquigarrow$ proper colourings

Motivation. Resource sharing: G is an *interference* graph

- Colouring a vertex assigns it a *transmission frequency*
- Adjacent vertices cannot use the same frequency
- Uncoloured vertices are *idle* (not transmitting)

Set-Up, Motivation and Examples

Set-Up. Let $G = (V, E)$ be a finite graph

- Our objective is to *properly colour* a (large) subset U of vertices
- If $u, u' \in U$ and $\{u, u'\} \in E$, then u and u' have different colours
- one colour $\leftrightsquigarrow$ independent sets, $U = V \leftrightsquigarrow$ proper colourings

Motivation. Resource sharing: G is an *interference* graph

- Colouring a vertex assigns it a *transmission frequency*
- Adjacent vertices cannot use the same frequency
- Uncoloured vertices are *idle* (not transmitting)

Examples. Frequency-limited communication

- Nearby users of short-range radio
- Fibreoptic routing (more on this later)

Model and Main Result

Glauber-Type Dynamics

Glauber-Type Dynamics

Let $K \in \mathbb{N}$ and $\Omega := \{\omega \in \{0, 1, \ldots, K\}^V \mid \omega \text{ proper}\}$, where

$$\omega \text{ is } \textit{proper} \quad \text{if} \quad \omega_u \neq \omega_v \quad \text{whenever} \quad \{u, v\} \in E \quad \text{and} \quad \omega_u + \omega_v > 0.$$

Glauber-Type Dynamics

Let $K \in \mathbb{N}$ and $\Omega := \{\omega \in \{0, 1, ..., K\}^V \mid \omega \text{ proper}\}$, where

$$\omega \text{ is } \textit{proper} \quad \text{if} \quad \omega_u \neq \omega_v \quad \text{whenever} \quad \{u, v\} \in E \quad \text{and} \quad \omega_u + \omega_v > 0.$$

Definition (Glauber-Type Dynamics: $\mathrm{MCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p})$)

- Select vertex $v \in V$ to update at rate λ_v, simultaneously over all vertices.

Glauber-Type Dynamics

Let $K \in \mathbb{N}$ and $\Omega := \{\omega \in \{0, 1, ..., K\}^V \mid \omega \text{ proper}\}$, where

$$\omega \text{ is } \textit{proper} \quad \text{if} \quad \omega_u \neq \omega_v \quad \text{whenever} \quad \{u, v\} \in E \quad \text{and} \quad \omega_u + \omega_v > 0.$$

Definition (Glauber-Type Dynamics: $\mathrm{MCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p})$)

- Select vertex $v \in V$ to update at rate λ_v, simultaneously over all vertices.
- When vertex $v \in V$ is chosen, toss a p_v-biassed coin: $C \sim \mathrm{Bern}(p_v)$.
  - If $C = 1$, choose a (non-zero) colour $k \in \{1, ..., K\}$ uniformly. If k is *available* for v, then paint v with colour k.
  - If $C = 0$, give it 'colour' 0—ie, *deactivate* it.

Glauber-Type Dynamics

Let $K \in \mathbb{N}$ and $\Omega := \{\omega \in \{0, 1, ..., K\}^V \mid \omega \text{ proper}\}$, where

ω is *proper* if $\omega_u \neq \omega_v$ whenever $\{u, v\} \in E$ and $\omega_u + \omega_v > 0$.

Definition (Glauber-Type Dynamics: $\mathrm{MCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p})$)

- Select vertex $v \in V$ to update at rate λ_v, simultaneously over all vertices.
- When vertex $v \in V$ is chosen, toss a p_v-biassed coin: $C \sim \mathrm{Bern}(p_v)$.
  - If $C = 1$, choose a (non-zero) colour $k \in \{1, ..., K\}$ uniformly. If k is *available* for v, then paint v with colour k.
  - If $C = 0$, give it 'colour' 0—ie, *deactivate* it.

Denote the equilibrium distribution π and the *equilibrium service rates*

$$s_v := \sum_{\omega \in \Omega : \omega_v \neq 0} \pi(\omega) \quad \text{for} \quad v \in V.$$

Glauber-Type Dynamics

Let $K \in \mathbb{N}$ and $\Omega := \{\omega \in \{0, 1, ..., K\}^V \mid \omega \text{ proper}\}$, where

ω is *proper* if $\omega_u \neq \omega_v$ whenever $\{u, v\} \in E$ and $\omega_u + \omega_v > 0$.

Definition (Glauber-Type Dynamics: $\mathrm{MCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p})$)

- Select vertex $v \in V$ to update at rate λ_v, simultaneously over all vertices.
- When vertex $v \in V$ is chosen, toss a p_v-biassed coin: $C \sim \mathrm{Bern}(p_v)$.
  - If $C = 1$, choose a (non-zero) colour $k \in \{1, ..., K\}$ uniformly. If k is *available* for v, then paint v with colour k.
  - If $C = 0$, give it 'colour' 0—ie, *deactivate* it.

Denote the equilibrium distribution π and the *equilibrium service rates*

$$s_v := \sum_{\omega \in \Omega : \omega_v \neq 0} \pi(\omega) \quad \text{for} \quad v \in V.$$

If preferred, at each discrete time-step, choose $v \in V$ with probability $\propto \lambda_v$.

Mixing Times

Mixing Times

Our main theorem establishes *fast mixing* of these Glauber-type dynamics.

Mixing Times

Our main theorem establishes *fast mixing* of these Glauber-type dynamics.

Definition (Mixing Times)

The TV distance between distributions μ and π on a set Ω is

$$\|\mu - \pi\|_{\mathsf{TV}} := \max_{A \subseteq \Omega} |\mu(A) - \pi(A)|.$$

Mixing Times

Our main theorem establishes *fast mixing* of these Glauber-type dynamics.

Definition (Mixing Times)

The TV distance between distributions μ and π on a set Ω is

$$\|\mu - \pi\|_{\mathsf{TV}} := \max_{A \subseteq \Omega} |\mu(A) - \pi(A)|.$$

The *mixing time* of a Markov chain $X = (X^t)_{t \ge 0}$ with equilibrium dist π is

$$t_{\mathsf{mix}}(\varepsilon) := \inf\{ t \ge 0 \mid \max_{x \in \Omega} \|\mathbb{P}_x[X^t \in \cdot] - \pi\| \le \varepsilon \} \quad \text{for} \quad \varepsilon \in (0,1).$$

Mixing Times

Our main theorem establishes *fast mixing* of these Glauber-type dynamics.

Definition (Mixing Times)

The TV distance between distributions μ and π on a set Ω is

$$\|\mu - \pi\|_{\mathsf{TV}} := \max_{A \subseteq \Omega} |\mu(A) - \pi(A)|.$$

The *mixing time* of a Markov chain $X = (X^t)_{t \geq 0}$ with equilibrium dist π is

$$t_{\mathsf{mix}}(\varepsilon) := \inf\{t \geq 0 \mid \max_{x \in \Omega} \|\mathbb{P}_x[X^t \in \cdot] - \pi\| \leq \varepsilon\} \quad \text{for} \quad \varepsilon \in (0,1).$$

Lemma (Coupling Method)

Let X and Y be a coupling of μ and π: ie, $X \sim \mu$ and $Y \sim \pi$. Then,

$$\|\mu - \pi\|_{TV} \leq \mathbb{P}[X \neq Y].$$

Mixing Times

Our main theorem establishes *fast mixing* of these Glauber-type dynamics.

Definition (Mixing Times)

The TV distance between distributions μ and π on a set Ω is

$$\|\mu - \pi\|_{\mathsf{TV}} \coloneqq \max_{A \subseteq \Omega} |\mu(A) - \pi(A)|.$$

The *mixing time* of a Markov chain $X = (X^t)_{t \geq 0}$ with equilibrium dist π is

$$t_{\mathsf{mix}}(\varepsilon) \coloneqq \inf\{t \geq 0 \mid \max_{x \in \Omega} \|\mathbb{P}_x[X^t \in \cdot] - \pi\| \leq \varepsilon\} \quad \text{for} \quad \varepsilon \in (0,1).$$

Lemma (Coupling Method)

Let X and Y be a coupling of μ and π: ie, $X \sim \mu$ and $Y \sim \pi$. Then,

$$\|\mu - \pi\|_{TV} \leq \mathbb{P}[X \neq Y].$$

Path coupling only requires contraction of adjacent configurations in expectation.

Main Result: Conditions for Fast Mixing

Main Result: Conditions for Fast Mixing

Theorem (Fast Mixing for MCH)

Let $n := |V|$ *and* $X \sim \mathsf{MCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p})$*. Suppose that there exists* $\beta > 0$ *such that*

$$\tfrac{1}{K} \textstyle\sum_{u \in V : \{u,v\} \in E} p_u \lambda_u / \lambda_v \le 1 - \beta \quad \text{for all} \quad v \in V.$$

Then,

$$\max_{x \in \Omega} \| \mathbb{P}_x[X^t \in \cdot] - \pi \|_{TV} \le \min\{2ne^{-\beta \lambda_{\min} t}, 1\} \quad \text{for all} \quad t \ge 0.$$

In particular, $t_{mix}(\varepsilon) \le (\beta \lambda_{\min})^{-1} \log(2n/\varepsilon) \quad \text{for all} \quad \varepsilon \in (0,1).$

Main Result: Conditions for Fast Mixing

Theorem (Fast Mixing for MCH)

Let $n := |V|$ *and* $X \sim \mathrm{MCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p})$*. Suppose that there exists* $\beta > 0$ *such that*

$$\tfrac{1}{K} \textstyle\sum_{u \in V : \{u,v\} \in E} p_u \lambda_u / \lambda_v \le 1 - \beta \quad \text{for all} \quad v \in V.$$

Then,

$$\max_{x \in \Omega} \| \mathbb{P}_x[X^t \in \cdot] - \pi \|_{TV} \le \min\{2 n e^{-\beta \lambda_{\min} t}, 1\} \quad \text{for all} \quad t \ge 0.$$

In particular, $t_{mix}(\varepsilon) \le (\beta \lambda_{\min})^{-1} \log(2n/\varepsilon)$ *for all* $\varepsilon \in (0,1)$.

Remark (Heuristic-Driven Choice of Parameters $(\boldsymbol{\lambda}, \boldsymbol{p})$)

The graph G and number K of colours are prescribed by the application.

- High-degree vertices have more impact, so update them faster: $\lambda_v \propto d_v$.
- If v is 'on' with prob p_v, then it removes $p_v d_v$ colour choices: $p_v d_v \propto K$.

Main Result: Conditions for Fast Mixing

Theorem (Fast Mixing for MCH)

Let $n := |V|$ *and* $X \sim \mathsf{MCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p})$. *Suppose that there exists* $\beta > 0$ *such that*

$$\tfrac{1}{K} \textstyle\sum_{u \in V : \{u,v\} \in E} p_u \lambda_u / \lambda_v \le 1 - \beta \quad \textit{for all} \quad v \in V.$$

Then,

$$\max_{x \in \Omega} \|\mathbb{P}_x[X^t \in \cdot] - \pi\|_{TV} \le \min\{2n e^{-\beta \lambda_{\min} t}, 1\} \quad \textit{for all} \quad t \ge 0.$$

In particular, $\quad t_{mix}(\varepsilon) \le (\beta \lambda_{\min})^{-1} \log(2n/\varepsilon) \quad$ *for all* $\quad \varepsilon \in (0,1)$.

Remark (Heuristic-Driven Choice of Parameters $(\boldsymbol{\lambda}, \boldsymbol{p})$)

The graph G and number K of colours are prescribed by the application.

- High-degree vertices have more impact, so update them faster: $\lambda_v \propto d_v$.
- If v is 'on' with prob p_v, then it removes $p_v d_v$ colour choices: $p_v d_v \propto K$.

The conditions are satisfied with $\beta = \frac{1}{3}$ if $\lambda_v = d_v / \bar{d}$ and $p_v \le \frac{2}{3} K / d_v$.
Further, under these conditions, $\frac{1}{3} p_v \le s_v \le p_v$ for all $v \in V$.

Application as a Queueing System

Motivation from Fibreoptic Networks

Motivation from Fibreoptic Networks

The original motivation for this system was to model **fibreoptic routing**.

Motivation from Fibreoptic Networks

The original motivation for this system was to model **fibreoptic routing**.

- Data in sent along cables between cities: eg, Bath $\rightarrow$ Reading $\rightarrow$ London
- A single frequency (colour) of light must be used for the *entire* route

Motivation from Fibreoptic Networks

The original motivation for this system was to model **fibreoptic routing**.

- Data in sent along cables between cities: eg, Bath $\to$ Reading $\to$ London
- A single frequency (colour) of light must be used for the *entire* route
- *Interfering routes* are ones which share a *link*, and must use *different* colours
- Cardiff $\to$ Bath $\to$ Reading interferes with Bath $\to$ Reading $\to$ London
- Routes which are physically far apart interfere, so cannot track everything

Motivation from Fibreoptic Networks

The original motivation for this system was to model **fibreoptic routing**.

- Data in sent along cables between cities: eg, Bath $\to$ Reading $\to$ London
- A single frequency (colour) of light must be used for the *entire* route
- *Interfering routes* are ones which share a *link*, and must use *different* colours
- Cardiff $\to$ Bath $\to$ Reading interferes with Bath $\to$ Reading $\to$ London
- Routes which are physically far apart interfere, so cannot track everything

Changing the colour involves re-encoding the data, which is a high overhead cost.

The MCH dynamics provide a decentralised way to share the resource.

Motivation from Fibreoptic Networks

The original motivation for this system was to model **fibreoptic routing**.

- Data in sent along cables between cities: eg, Bath $\to$ Reading $\to$ London
- A single frequency (colour) of light must be used for the *entire* route
- *Interfering routes* are ones which share a *link*, and must use *different* colours
- Cardiff $\to$ Bath $\to$ Reading interferes with Bath $\to$ Reading $\to$ London
- Routes which are physically far apart interfere, so cannot track everything

Changing the colour involves re-encoding the data, which is a high overhead cost.

The MCH dynamics provide a decentralised way to share the resource.

- It is fast and easy to test if a given colour is available along the route
- *Optical switches* can be configured rapidly to set-up a *light path*
- The particular light path is used to transmit data until it is refreshed

Queueing Network

Queueing Network

Definition (MCH Queueing Network: $\mathrm{QMCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p}; \boldsymbol{\nu}, \boldsymbol{\mu})$)

Let $X \sim \mathrm{MCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p})$. The state space of Q is $\mathbb{N}^V$.

Queueing Network

Definition (MCH Queueing Network: $\mathrm{QMCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p}; \boldsymbol{\nu}, \boldsymbol{\mu})$)

Let $X \sim \mathrm{MCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p})$. The state space of Q is $\mathbb{N}^V$. Its transition rates are

$$q \to \begin{cases} q^{v,+} & \text{at rate} \quad \nu_v \\ q^{v,-} & \text{at rate} \quad \mu_v \mathbf{1}\{x_v \neq 0\} \end{cases} \quad \begin{array}{l} \text{where} \quad q_u^{v,\pm} := q_u \pm \mathbf{1}\{u = v\} \\ \text{given MCH configuration } X = x \end{array}$$

In other words, *jobs* arrive to vertex $v \in V$ at rate ν_v and are processed at rate μ_v *provided* vertex v is *active* in the underlying MCH configuration (ie, $x_v \neq 0$).

Queueing Network

Definition (MCH Queueing Network: $\text{QMCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p}; \boldsymbol{\nu}, \boldsymbol{\mu})$)

Let $X \sim \text{MCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p})$. The state space of Q is $\mathbb{N}^V$. Its transition rates are

$$q \to \begin{cases} q^{v,+} & \text{at rate} \quad \nu_v \\ q^{v,-} & \text{at rate} \quad \mu_v \mathbf{1}\{x_v \neq 0\} \end{cases} \quad \begin{array}{l} \text{where} \quad q_u^{v,\pm} := q_u \pm \mathbf{1}\{u = v\} \\ \text{given MCH configuration } X = x \end{array}$$

In other words, *jobs* arrive to vertex $v \in V$ at rate ν_v and are processed at rate μ_v *provided* vertex v is *active* in the underlying MCH configuration (ie, $x_v \neq 0$).

Theorem

Suppose $\lambda_v = d_v / \bar{d}$, $p_v \leq \frac{2}{3} K / d_v$ *and* $\nu_v < \frac{1}{3} p_v$ *for all* v.

Queueing Network

Definition (MCH Queueing Network: $\mathrm{QMCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p}; \boldsymbol{\nu}, \boldsymbol{\mu})$)

Let $X \sim \mathrm{MCH}_\Omega(\boldsymbol{\lambda}, \boldsymbol{p})$. The state space of Q is $\mathbb{N}^V$. Its transition rates are

$$q \to \begin{cases} q^{v,+} & \text{at rate} \quad \nu_v \\ q^{v,-} & \text{at rate} \quad \mu_v \mathbf{1}\{x_v \neq 0\} \end{cases} \quad \begin{array}{l} \text{where} \quad q_u^{v,\pm} := q_u \pm \mathbf{1}\{u = v\} \\ \text{given MCH configuration } X = x \end{array}$$

In other words, *jobs* arrive to vertex $v \in V$ at rate ν_v and are processed at rate μ_v *provided* vertex v is *active* in the underlying MCH configuration (ie, $x_v \neq 0$).

Theorem

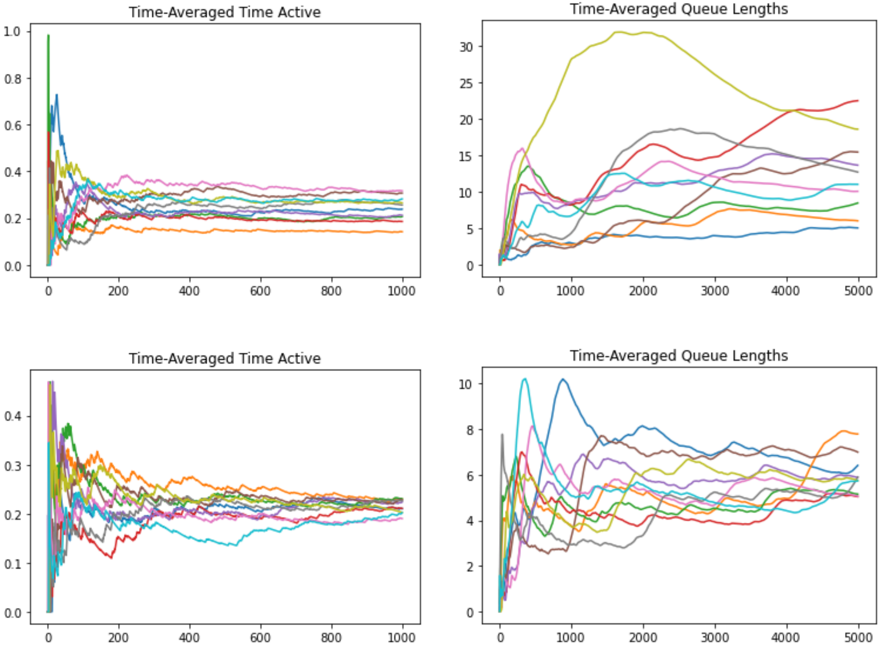
Suppose $\lambda_v = d_v/\bar{d}$, $p_v \le \frac{2}{3}K/d_v$ and $\nu_v < \frac{1}{3}p_v$ for all v. Then, in equilibrium,

$$\mathbb{E}[Q_v^0] \le \frac{18(\bar{d}/d_{\min}) n \log(2n/e)}{(s_v - \nu_v)^2} \quad \textit{for all} \quad v \in V.$$

Simulations

Graphs: Erdős–Rényi (top) and random-regular (bottom) graphs

- $n = 500$ vertices and average degree $\bar{d} = 40$
- $K = 10$ colours and probabilities $p_v = \frac{4}{5} eK/d_v \approx 0.5$


Time-Averaged Time Active
1.0
0.8
0.6
0.4
0.2
0.0
0
200
400
600
800
1000
Time-Averaged Queue Lengths
30
25
20
15
10
5
0
0
1000
2000
3000
4000
5000
Time-Averaged Time Active
0.4
0.3
0.2
0.1
0.0
0
200
400
600
800
1000
Time-Averaged Queue Lengths
10
8
6
4
2
0
0
1000
2000
3000
4000
5000