Advanced Monte Carlo Methods

P. Del Moral

INRIA Bordeaux

Nelder Fellow Lecture Series, March 2024

Part I - Introduction - From Meta-heuristics to Probabilistic models

Outline

Boltzmann-Gibbs measures

$$\pi(dx) := \frac{1}{\mathcal{Z}_\beta}\ e^{-\beta\ \ U(x)}\ \nu(dx) \quad \text{or} \quad \pi(dx) := \frac{1}{\mathcal{Z}_A}\ 1_A(x)\ \nu(dx)$$

Boltzmann-Gibbs measures

$$\pi(dx) := \frac{1}{\mathcal{Z}_\beta}\, e^{-\beta\ U(x)}\ \nu(dx) \quad \text{or} \quad \pi(dx) := \frac{1}{\mathcal{Z}_A}\ 1_A(x)\ \nu(dx)$$

Some examples:

- *Physics: Ising/Sherrington-Kirkpatrick model*:
  - State space:
    $$x \in \{-1,+1\}^{\{1,\ldots,L\}^2} \text{ with } \lambda(x) = 2^{-L^2}$$
  - Potential/Energy function
    $$U(x) = h\sum_i x(i) - J\sum_{i\sim j} \theta_{i,j}\ x(i)\ x(j)$$

Computer Science/Op. Research

Ex.: Traveling Salesman m cities $e_1, \ldots, e_m \rightsquigarrow$ State space:

$$x \in \mathcal{G}_m \quad \text{with counting measure} \quad \nu(dx) = \frac{1}{m!}$$

$\rightsquigarrow$ *Potential/Energy function (convention* $x(m+1) = x(1)$*)*

$$U(x) = \sum_{p=1}^{m} d(e_{x(p)}, e_{x(p+1)})$$

Computer Science/Op. Research

Ex.: Traveling Salesman m cities $e_1, \ldots, e_m$ $\rightsquigarrow$ State space:

$$x \in \mathcal{G}_m \quad \text{with counting measure} \quad \nu(dx) = \frac{1}{m!}$$

$\rightsquigarrow$ *Potential/Energy function (convention $x(m+1) = x(1)$)*

$$U(x) = \sum_{p=1}^{m} d(e_{x(p)}, e_{x(p+1)})$$

Important observation: $U_\star = \min U$

$$\frac{1}{\sum_y e^{-\beta\ (U(x)-U_\star)}}\ e^{-\beta\ (U(x)-U_\star)} \simeq_{\beta\uparrow\infty} \frac{1}{\#\{y\ :\ U(y) = U_\star\}}\ 1_{U(x)=U_\star}$$

$\rightsquigarrow$ **Fundamental Principle (for proba/stats/physics/...)**

Global optimization $\Leftrightarrow$ Sampling BG measures with large β

Engineering: Black Box/Inverse problems

$$\text{Inputs} \;=\; X \;\;\rightarrow \boxed{\text{Numerical codes}\;\;\mathsf{F}} \rightarrow\;\; \text{Outputs} \;=\; Y = F(X)$$

- **State space** = **inputs** $x \in S \oplus$ **some distribution** $\lambda = \mathrm{Law}(X)$

Engineering: Black Box/Inverse problems

$$\text{Inputs} = X \rightarrow \boxed{\text{Numerical codes } \mathsf{F}} \rightarrow \text{Outputs} = Y = F(X)$$

- **State space = inputs** $x \in S \oplus$ **some distribution** $\lambda = \mathrm{Law}(X)$
- **Potential/Energy function** $\sim B := F(A)$ **critical level set**

$$e^{-\beta U(x)} \overset{ex.\ U=1_{A^c}}{\simeq}_{\beta\uparrow} 1_A(x) \Rightarrow \pi = \mathrm{Law}(X \mid X \in A)$$

Engineering: Black Box/Inverse problems

$$\text{Inputs} \; = \; X \;\; \rightarrow \boxed{\text{Numerical codes} \;\; \mathsf{F}} \rightarrow \;\; \text{Outputs} \; = \; Y = F(X)$$

- **State space = inputs** $x \in S \oplus$ **some distribution** $\lambda = \mathrm{Law}(X)$
- **Potential/Energy function** $\sim B := F(A)$ **critical level set**

$$e^{-\beta U(x)} \overset{ex.\; U=1_{A^c}}{\simeq}_{\beta\uparrow} 1_A(x) \;\Rightarrow\; \pi = \mathrm{Law}(X \mid X \in A)$$

$\oplus$ **Normalizing cts :** $\mathbb{P}(X \in A)$ **= default/critical probab,. . .**

Objectives

- **Compute/Estimate normalizing constant**

Objectives

- **Compute/Estimate normalizing constant**
- **Sampling according to target distribution π**

2 Heuristic random search/sampling options

2 Heuristic random search/sampling options

Option 1: Sequentially : $X_0 \rightsquigarrow X_1 \rightsquigarrow \ldots$ **s.t.** X_k **"almost iid"** $\sim \pi$

$$\frac{1}{\mathbf{n}} \sum_{0 \leq k < \mathbf{n}} f(X_k) \simeq_{n \uparrow \infty} \int f(x)\, \pi(dx)$$

2 Heuristic random search/sampling options

Option 1: **Sequentially :** $X_0 \rightsquigarrow X_1 \rightsquigarrow \ldots$ **s.t.** X_k **"almost iid"** $\sim \pi$

$$\frac{1}{\mathbf{n}} \sum_{0 \leq k < \mathbf{n}} f(X_k) \simeq_{n \uparrow \infty} \int f(x)\, \pi(dx)$$

$\Longleftrightarrow$ **Shaking process i.e.**

$$X_0 \sim \pi \rightsquigarrow X_1 \sim \pi$$

2 Heuristic random search/sampling options

Option 1: **Sequentially :** $X_0 \rightsquigarrow X_1 \rightsquigarrow \ldots$ **s.t.** X_k **"almost iid"** $\sim \pi$

$$\frac{1}{\mathbf{n}} \sum_{0 \leq k < \mathbf{n}} f(X_k) \simeq_{n \uparrow \infty} \int f(x)\, \pi(dx)$$

$\Longleftrightarrow$ **Shaking process i.e.**

$$X_0 \sim \pi \rightsquigarrow X_1 \sim \pi$$

NOTE:

n = precision parameter = Computational power = time horizon/nb runs =...

Ex.: Boltzmann-Gibbs targets

$$\pi_\beta(dx) := \frac{1}{\mathcal{Z}_\beta}\; e^{-\beta\ U(x)} \underbrace{\nu(dx)}_{\text{ex.: } \mathcal{N}(0,1), Lebesgue, counting, volume, \ldots}$$

First ingredient

$\rightsquigarrow$ (To simplify) Choose a simple/feasible+reversible local exploration of the solution space:

Ex.: Boltzmann-Gibbs targets

$$\pi_\beta(dx) := \frac{1}{\mathcal{Z}_\beta}\ e^{-\beta\ U(x)} \quad \underbrace{\nu(dx)}_{\text{ex.: } \mathcal{N}(0,1),\, Lebesgue,\, counting,\, volume,\, \ldots}$$

First ingredient

$\rightsquigarrow$ (To simplify) Choose a simple/feasible+reversible local exploration of the solution space:

$\Longleftrightarrow$ A way of **exploring locally** $x \rightsquigarrow y$ the solution space s.t.

$$\nu(dx) \times K(x, dy) = \nu(dy) \times K(y, dx)$$

Ex.: Boltzmann-Gibbs targets

$$\pi_\beta(dx) := \frac{1}{\mathcal{Z}_\beta}\ e^{-\beta\ U(x)}\ \underbrace{\nu(dx)}_{\text{ex.: } \mathcal{N}(0,1), Lebesgue, counting, volume, \ldots}$$

First ingredient

$\rightsquigarrow$ (To simplify) Choose a simple/feasible+reversible local exploration of the solution space:

$\Longleftrightarrow$ A way of **exploring locally** $x \rightsquigarrow y$ the solution space s.t.

$$\nu(dx) \times K(x, dy) = \nu(dy) \times K(y, dx)$$

Important example $\nu(dx) \propto e^{-x^2/2}dx$

$$x \rightsquigarrow y = \sqrt{1-\epsilon}\ x + \sqrt{\epsilon}\ W \quad \text{with an } \mathcal{N}(0,1)\text{-sample } W$$

Proba. design of "π_β-shakers" $\rightsquigarrow$ 2 steps random search

Accept/Reject local moves:

$$\mathbf{x} \overset{K}{\rightsquigarrow} \mathbf{y} \rightsquigarrow \mathbf{z} = \begin{cases} y \text{ if } U(y) \leq U(x) \\ \text{otherwise} \\ z' = \begin{cases} y & \text{with proba} \quad e^{-\beta\ (U(y)-U(x))} \\ x & \text{with proba} \quad 1 - e^{-\beta\ (U(y)-U(x))} \end{cases} \end{cases}$$

Proba. design of "π_β-shakers" $\rightsquigarrow$ 2 steps random search

Accept/Reject local moves:

$$\mathbf{x} \overset{K}{\rightsquigarrow} \mathbf{y} \rightsquigarrow \mathbf{z} = \begin{cases} y \text{ if } U(y) \leq U(x) \\ \text{otherwise} \\ z' = \begin{cases} y & \text{with proba} \quad e^{-\beta\,(U(y)-U(x))} \\ x & \text{with proba} \quad 1 - e^{-\beta\,(U(y)-U(x))} \end{cases} \end{cases}$$

Non homogeneous version = Simulated Annealing ⊂ "Meta-heuristic"

Simulated Annealing = **SA** $\subset$ **"Meta-heuristic"**:

- "The simplest and best-known meta-heuristic method for addressing difficult black box global optimization problems. It is massively used on real-life applications..." Handbook of Metaheuristics (19)

Simulated Annealing = **SA** $\subset$ **"Meta-heuristic"**:

- "The simplest and best-known meta-heuristic method for addressing difficult black box global optimization problems. It is massively used on real-life applications. . ." Handbook of Metaheuristics (19)
- " A meta-heuristic to approximate global optimization. " (Wikipedia)

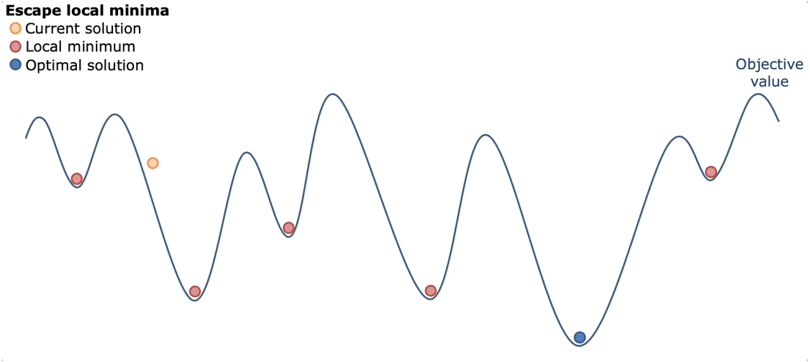
Simulated Annealing = **SA** $\subset$ **"Meta-heuristic"**:

- "The simplest and best-known meta-heuristic method for addressing difficult black box global optimization problems. It is massively used on real-life applications..." Handbook of Metaheuristics (19)
- " A meta-heuristic to approximate global optimization. " (Wikipedia)
- "A heuristic solution generation process that relies on logic and rules. " Encyclopedia of Ecology (08)

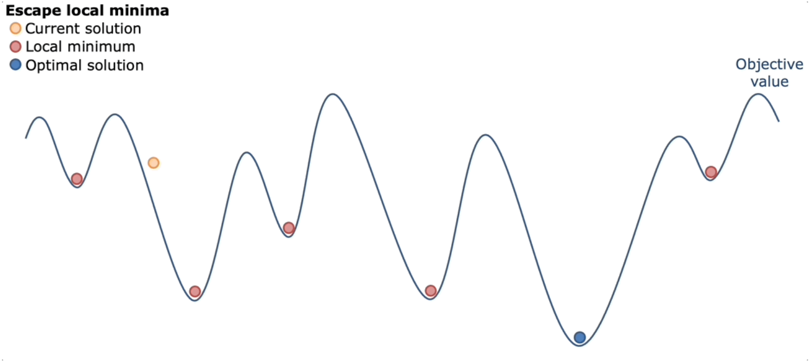
Simulated Annealing = **SA** $\subset$ **"Meta-heuristic"**:

- "The simplest and best-known meta-heuristic method for addressing difficult black box global optimization problems. It is massively used on real-life applications..." Handbook of Metaheuristics (19)
- " A meta-heuristic to approximate global optimization. " (Wikipedia)
- "A heuristic solution generation process that relies on logic and rules. " Encyclopedia of Ecology (08)
- ... $\rightsquigarrow$ *some maths on generalized SA* (LSP-Preprint1996/SIAM1999)

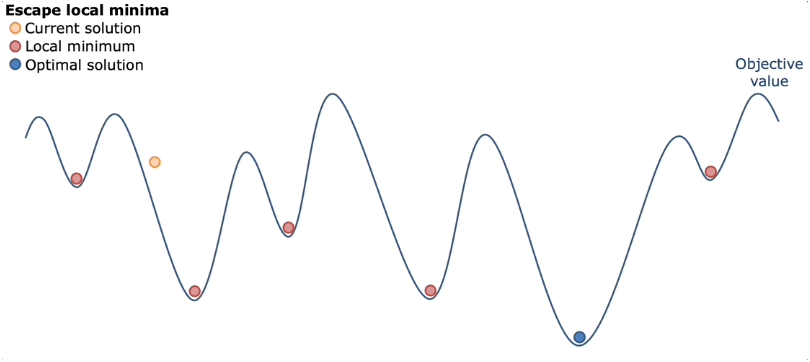
SA Numerical proof - Quod erat demonstratum



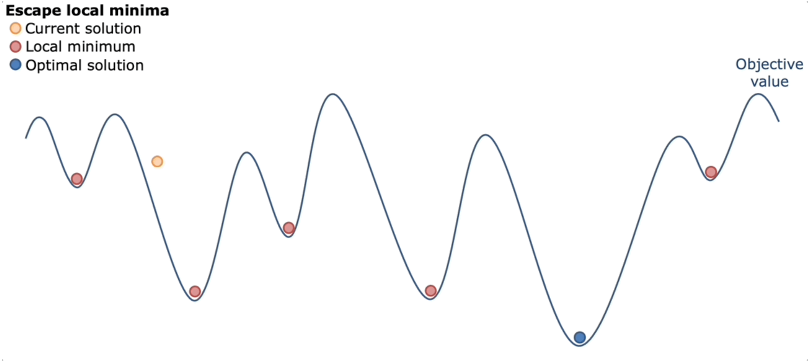
SA Numerical proof - Quod erat demonstratum



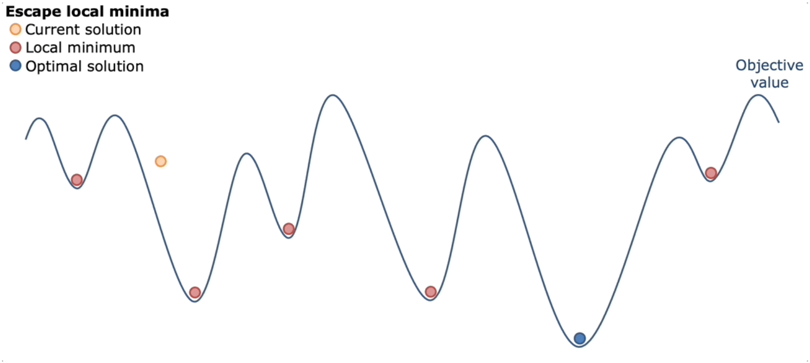
SA Numerical proof - Quod erat demonstratum



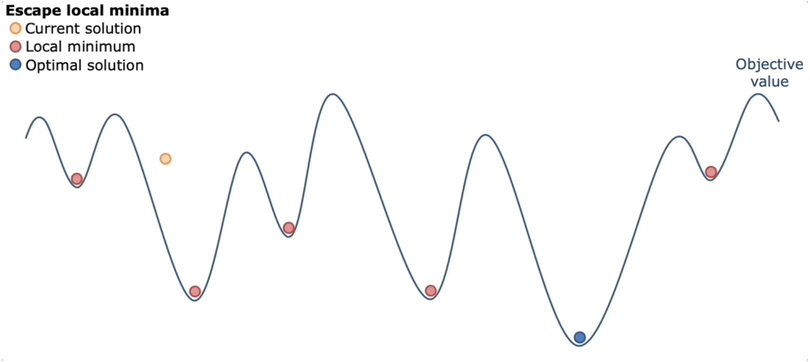
SA Numerical proof - Quod erat demonstratum



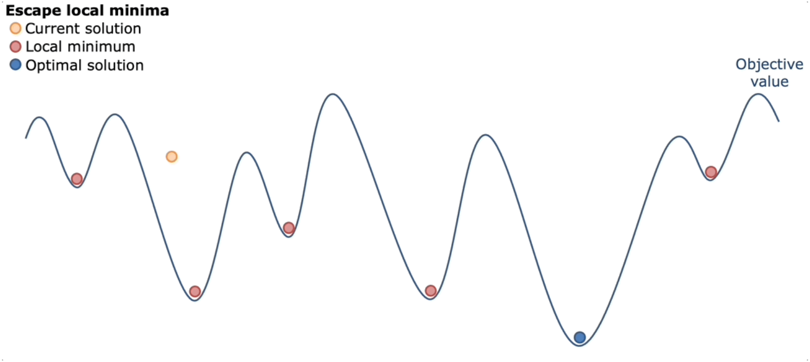
SA Numerical proof - Quod erat demonstratum



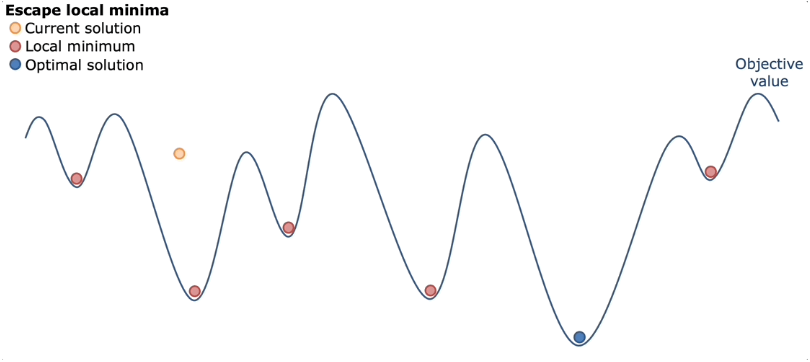
SA Numerical proof - Quod erat demonstratum



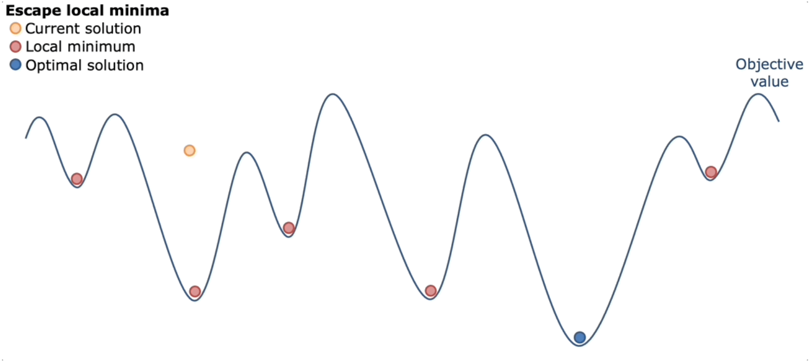
SA Numerical proof - Quod erat demonstratum



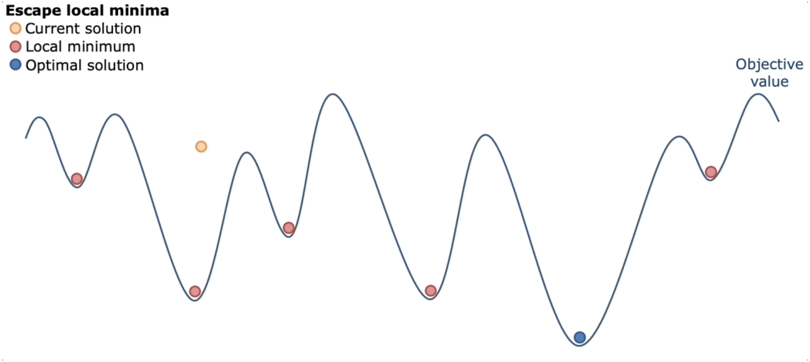
SA Numerical proof - Quod erat demonstratum



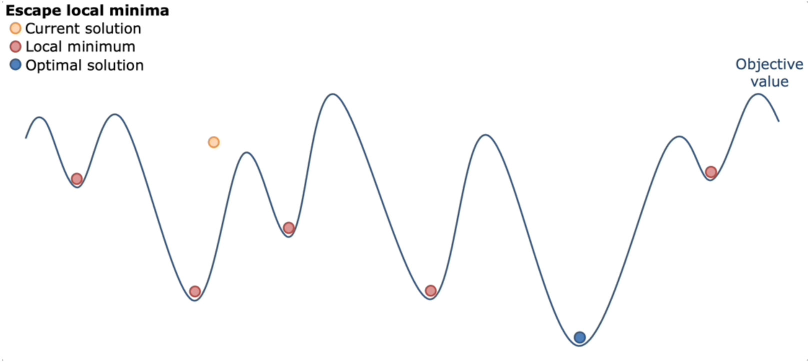
SA Numerical proof - Quod erat demonstratum



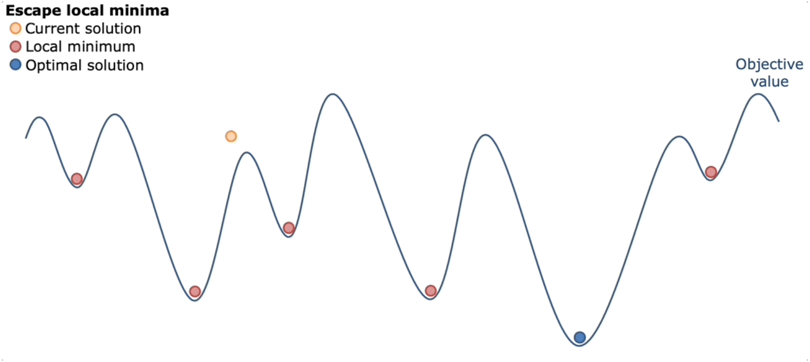
SA Numerical proof - Quod erat demonstratum



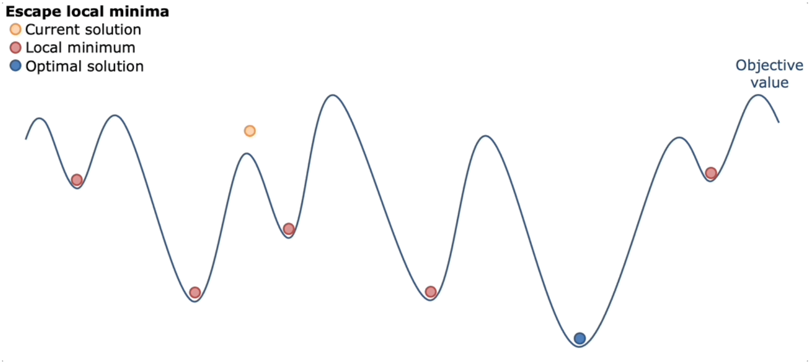
SA Numerical proof - Quod erat demonstratum



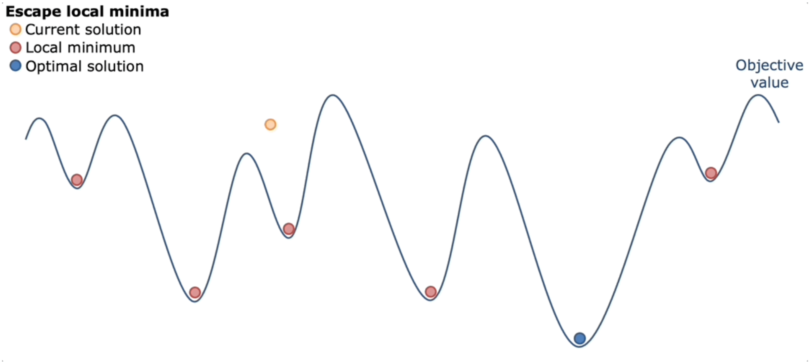
SA Numerical proof - Quod erat demonstratum



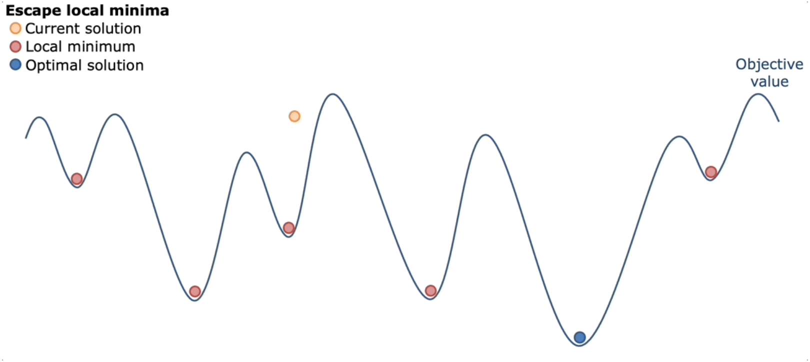
SA Numerical proof - Quod erat demonstratum



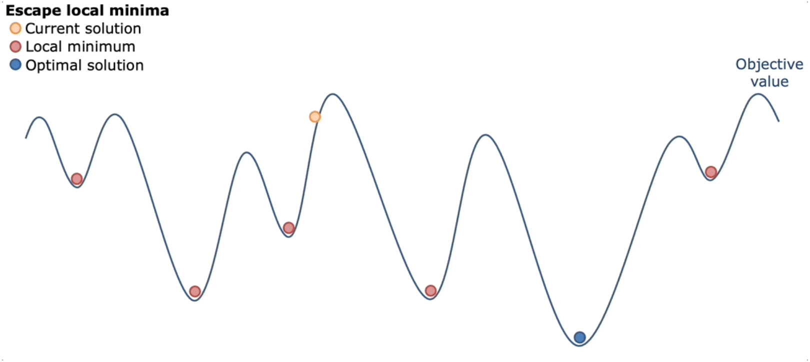
SA Numerical proof - Quod erat demonstratum



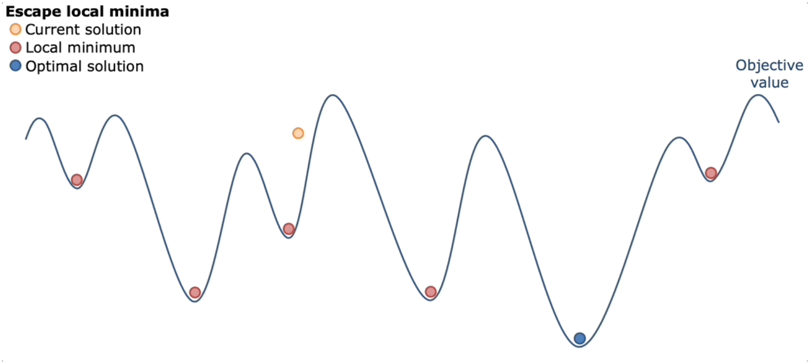
SA Numerical proof - Quod erat demonstratum



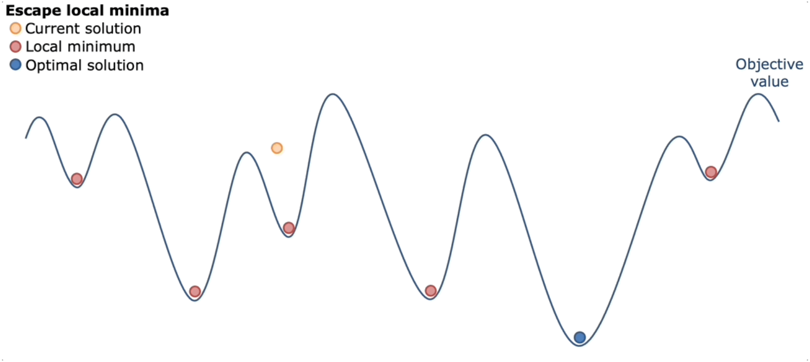
SA Numerical proof - Quod erat demonstratum



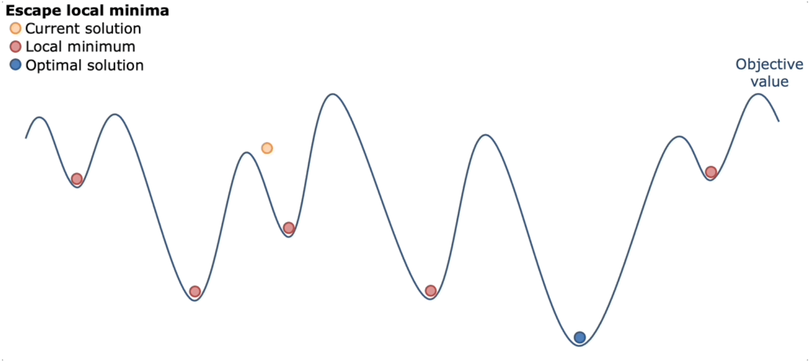
SA Numerical proof - Quod erat demonstratum



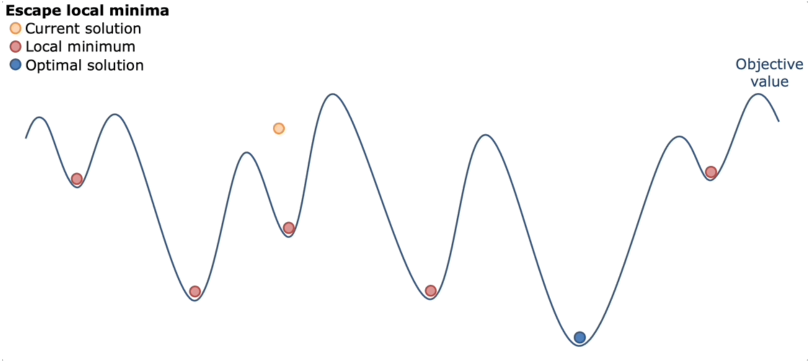
SA Numerical proof - Quod erat demonstratum



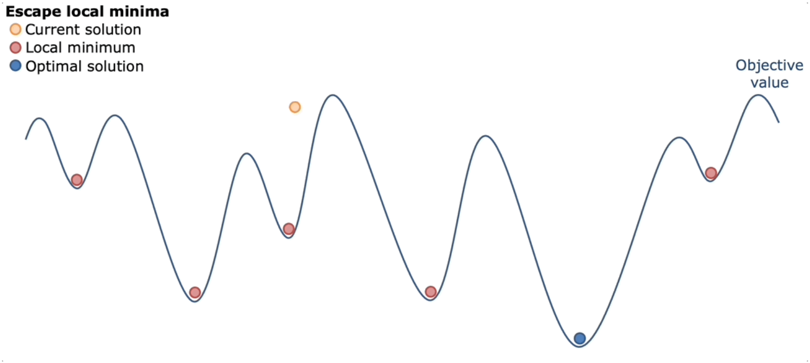
SA Numerical proof - Quod erat demonstratum



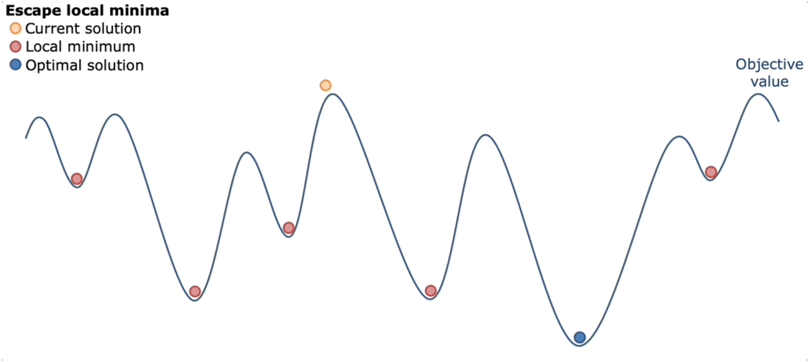
SA Numerical proof - Quod erat demonstratum



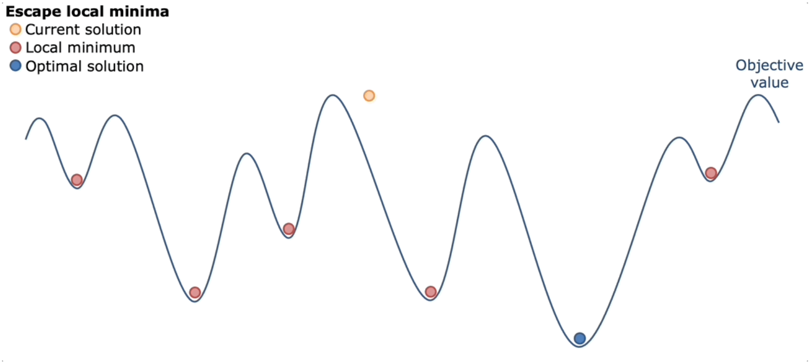
SA Numerical proof - Quod erat demonstratum



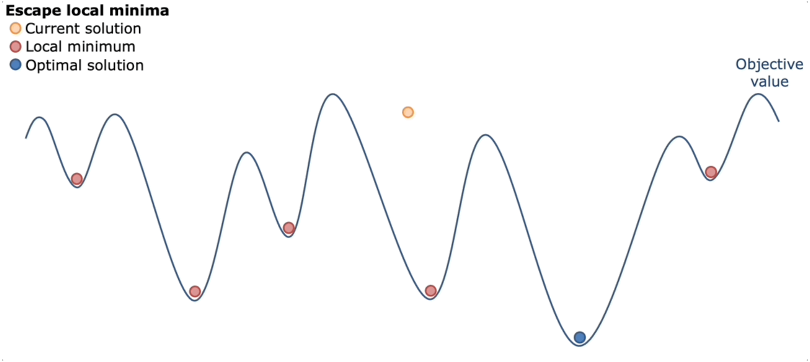
SA Numerical proof - Quod erat demonstratum



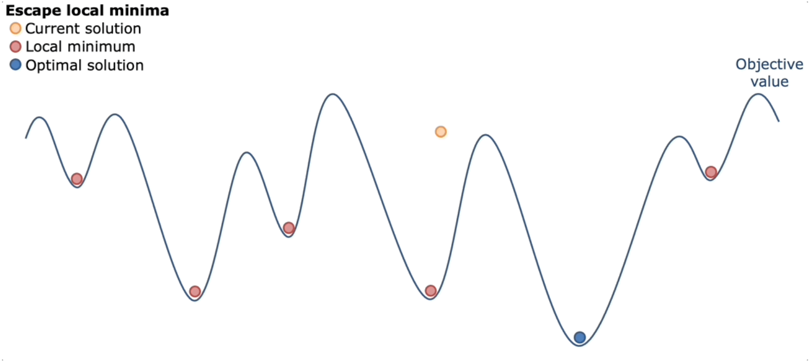
SA Numerical proof - Quod erat demonstratum



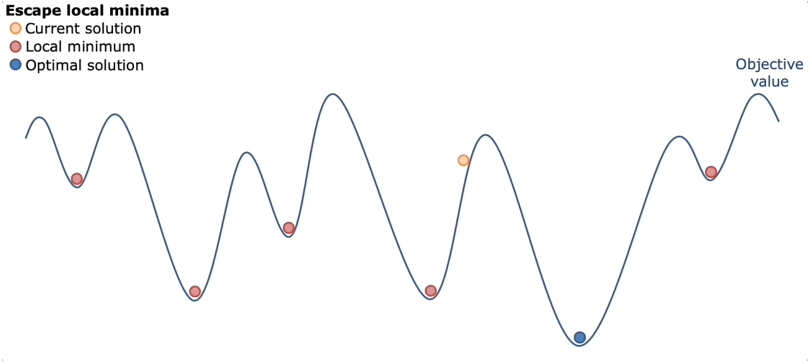
SA Numerical proof - Quod erat demonstratum



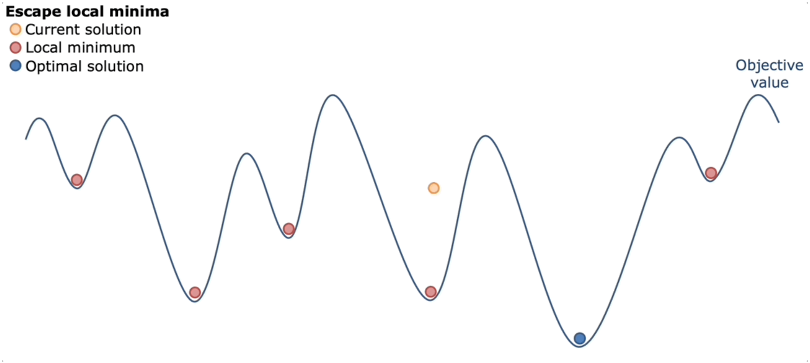
SA Numerical proof - Quod erat demonstratum



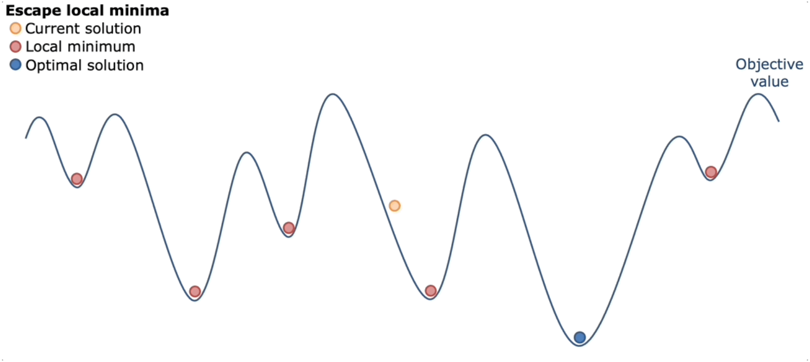
SA Numerical proof - Quod erat demonstratum



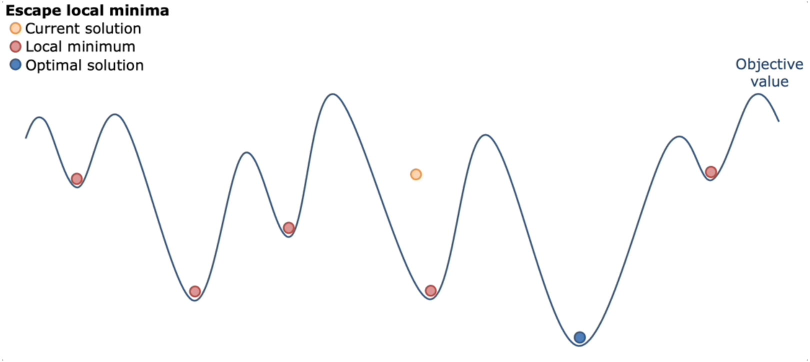
SA Numerical proof - Quod erat demonstratum



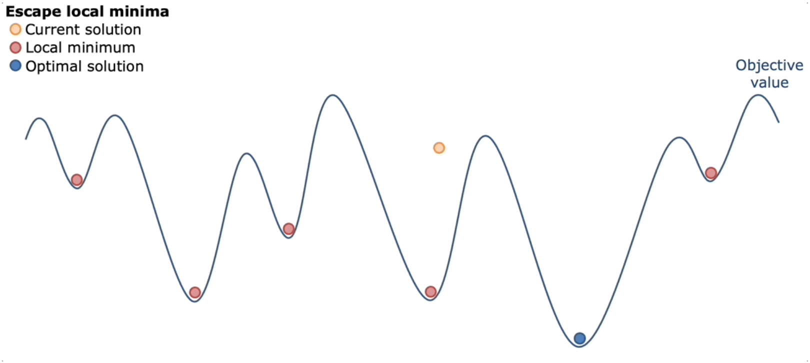
SA Numerical proof - Quod erat demonstratum



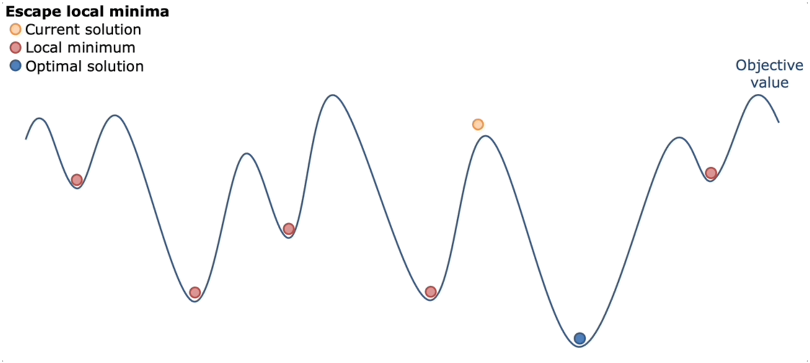
SA Numerical proof - Quod erat demonstratum



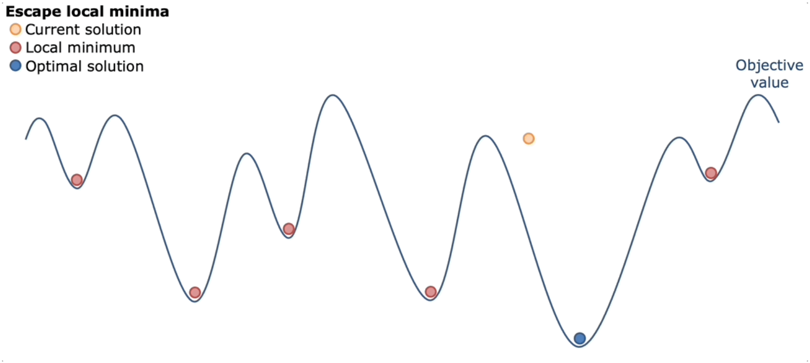
SA Numerical proof - Quod erat demonstratum



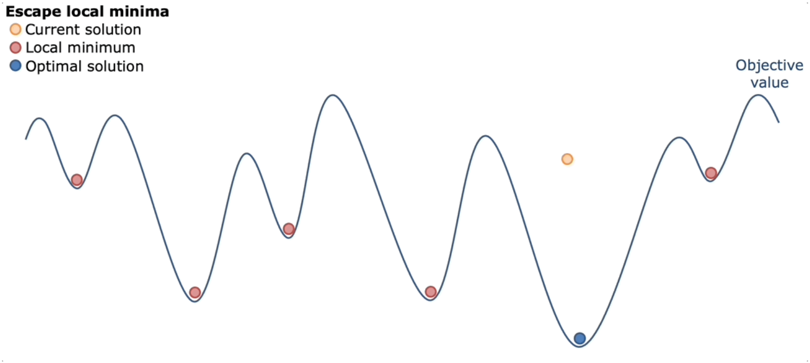
SA Numerical proof - Quod erat demonstratum



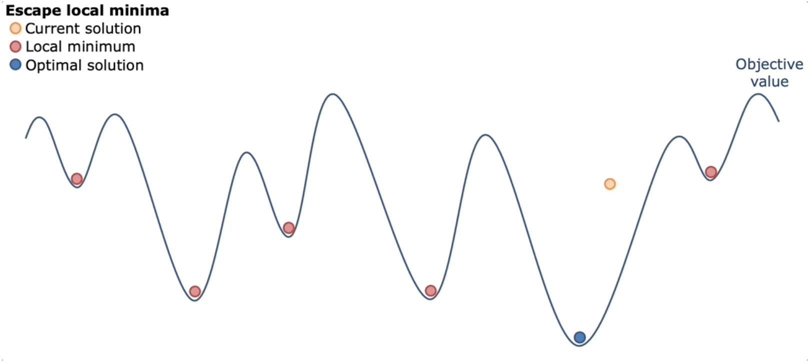
SA Numerical proof - Quod erat demonstratum



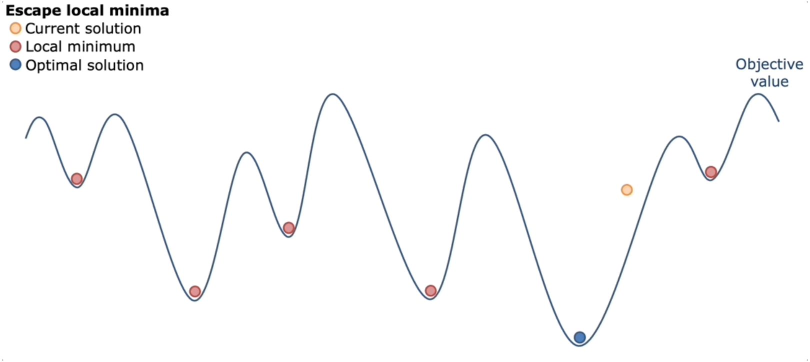
SA Numerical proof - Quod erat demonstratum



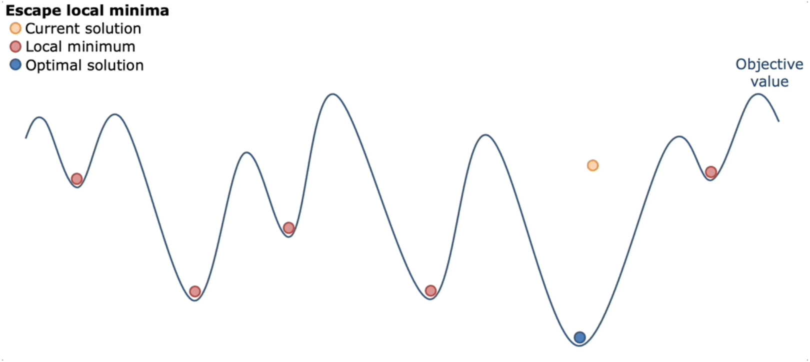
SA Numerical proof - Quod erat demonstratum



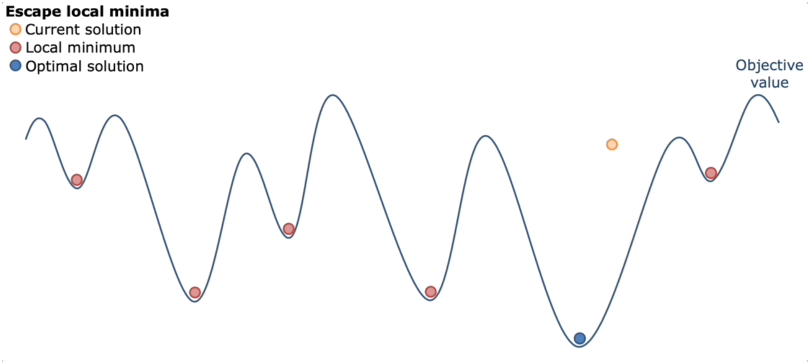
SA Numerical proof - Quod erat demonstratum



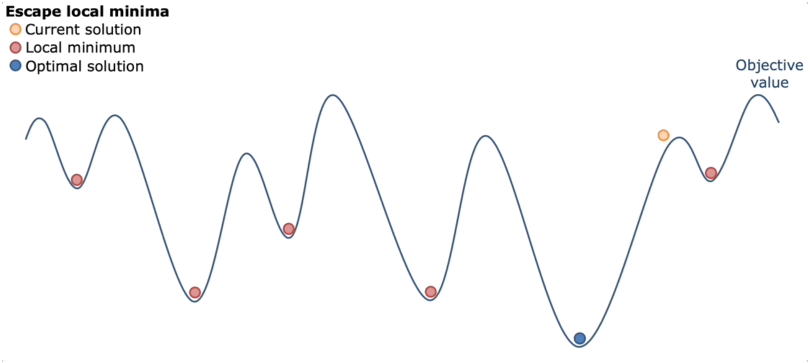
SA Numerical proof - Quod erat demonstratum



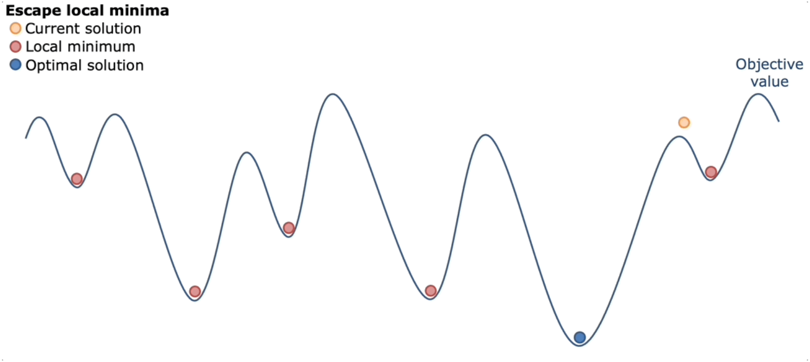
SA Numerical proof - Quod erat demonstratum



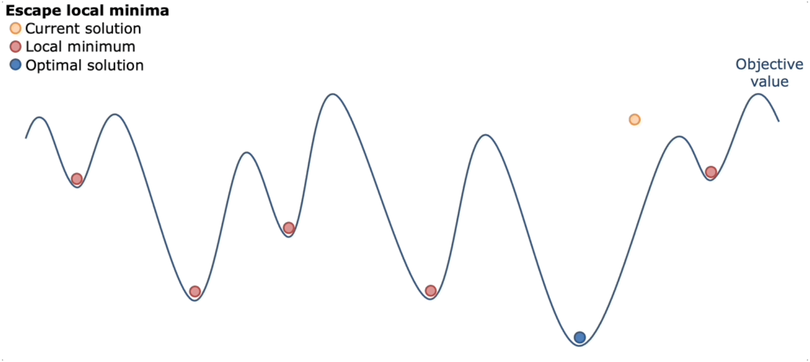
SA Numerical proof - Quod erat demonstratum



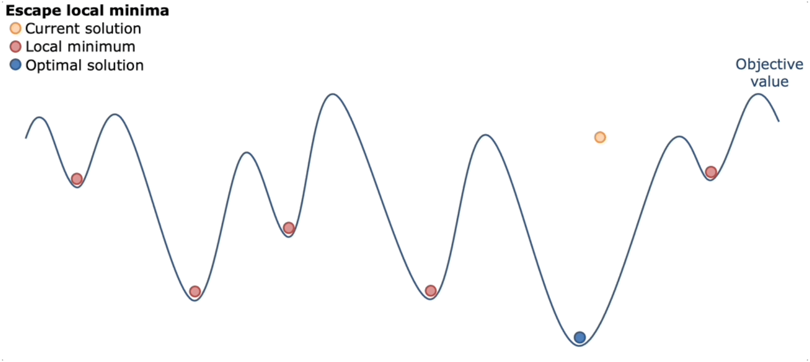
SA Numerical proof - Quod erat demonstratum



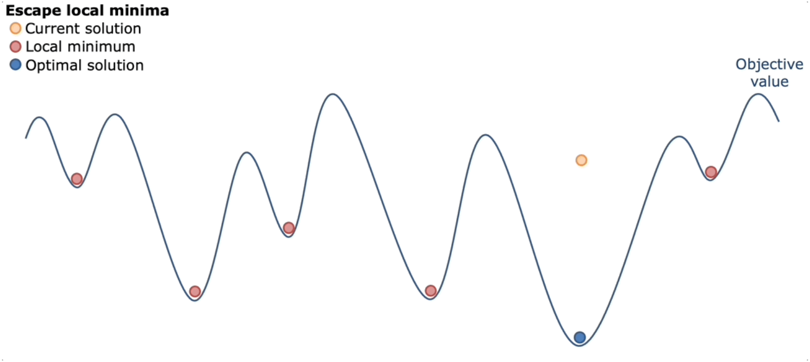
SA Numerical proof - Quod erat demonstratum



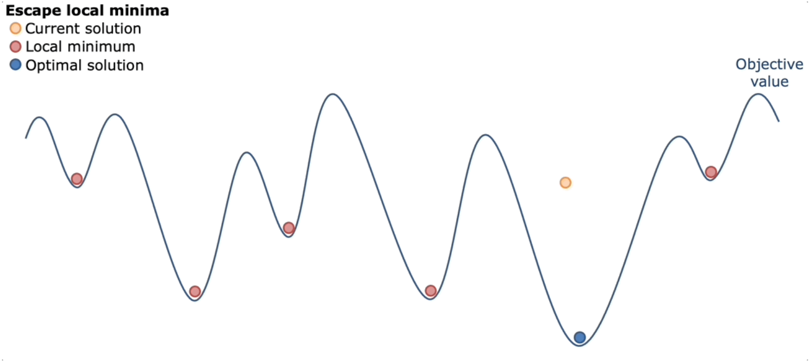
SA Numerical proof - Quod erat demonstratum



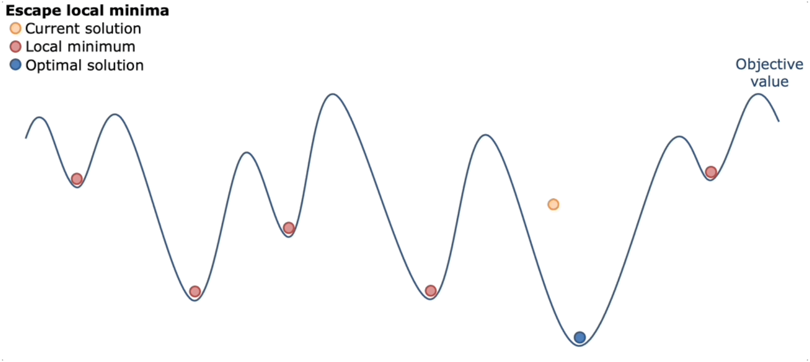
SA Numerical proof - Quod erat demonstratum



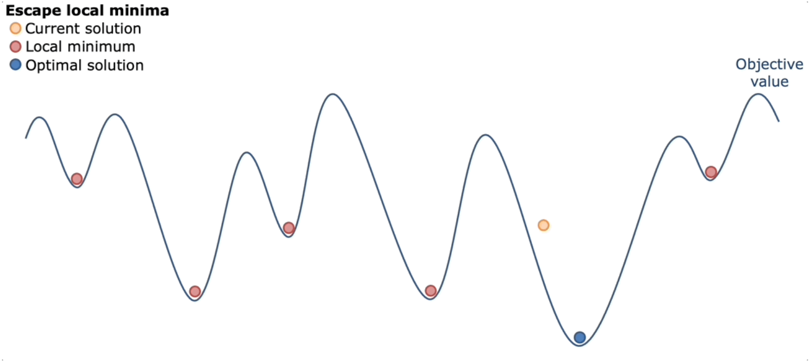
SA Numerical proof - Quod erat demonstratum



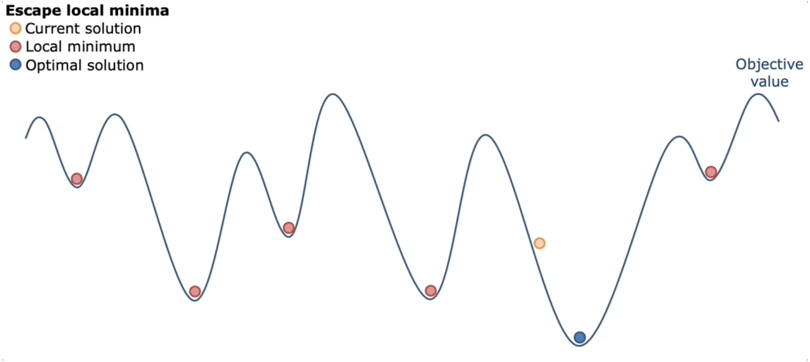
SA Numerical proof - Quod erat demonstratum



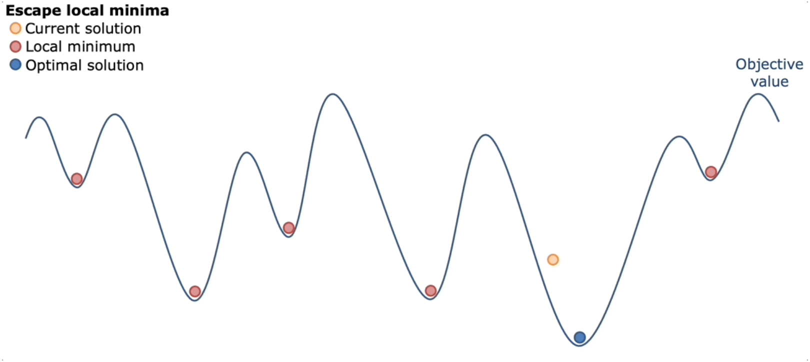
SA Numerical proof - Quod erat demonstratum



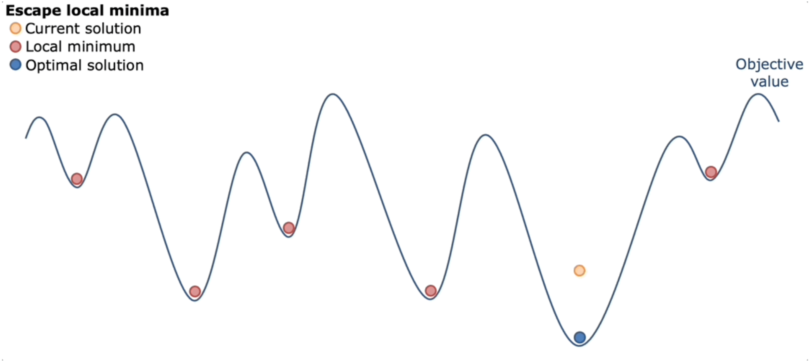
SA Numerical proof - Quod erat demonstratum



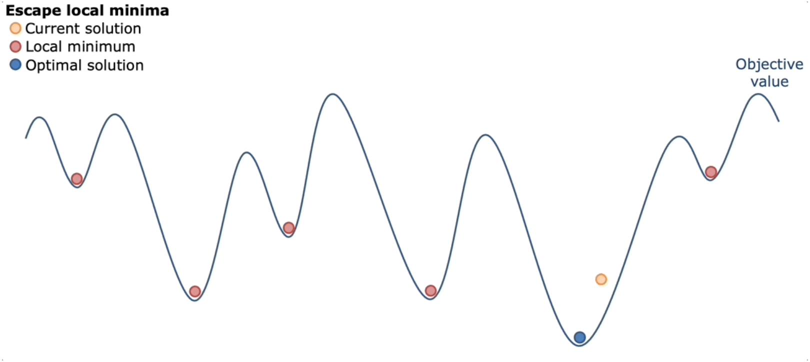
SA Numerical proof - Quod erat demonstratum



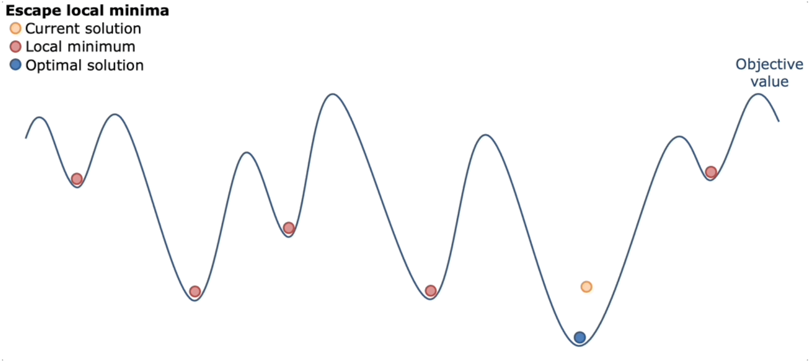
SA Numerical proof - Quod erat demonstratum



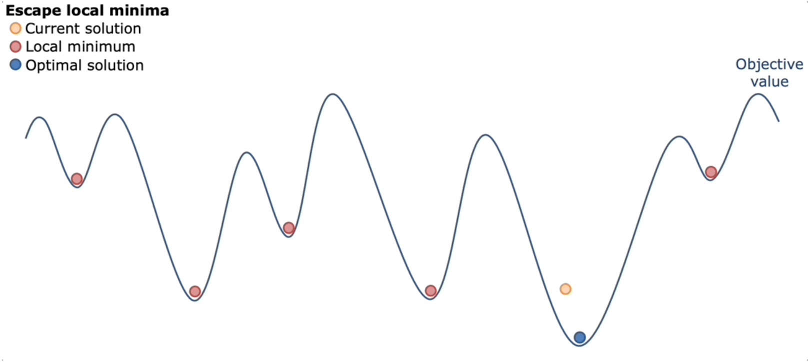
SA Numerical proof - Quod erat demonstratum



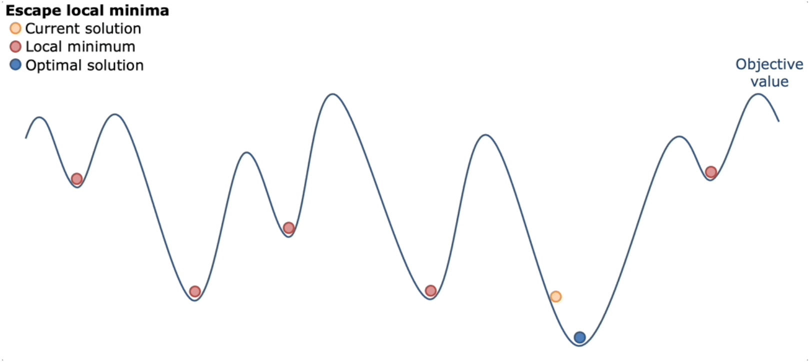
SA Numerical proof - Quod erat demonstratum



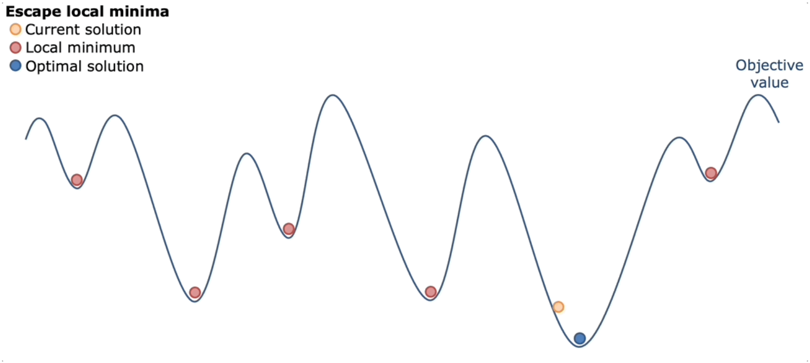
SA Numerical proof - Quod erat demonstratum



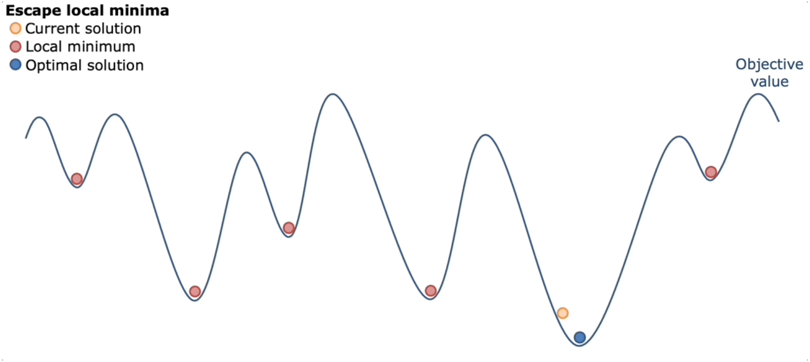
SA Numerical proof - Quod erat demonstratum



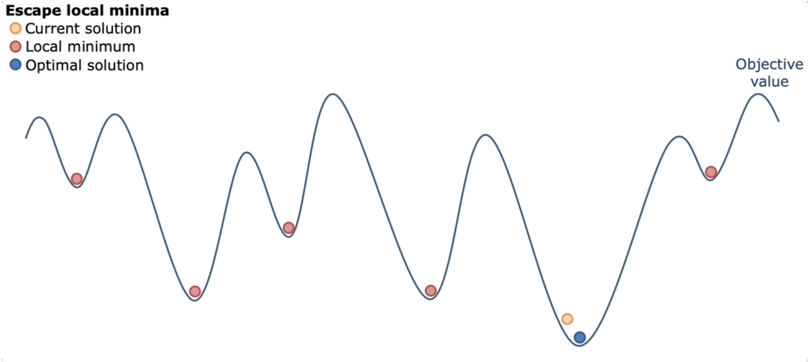
SA Numerical proof - Quod erat demonstratum



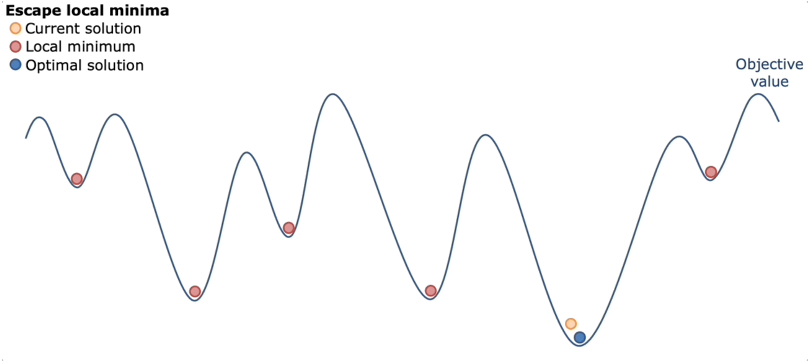
SA Numerical proof - Quod erat demonstratum



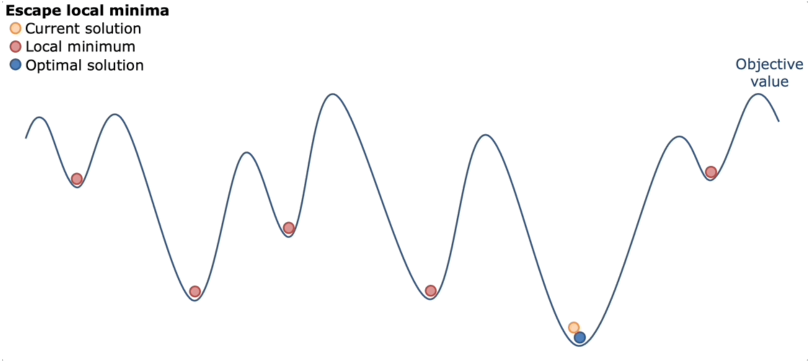
SA Numerical proof - Quod erat demonstratum



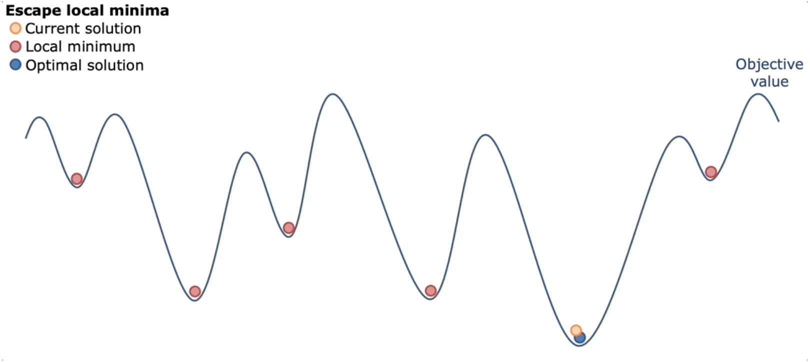
SA Numerical proof - Quod erat demonstratum



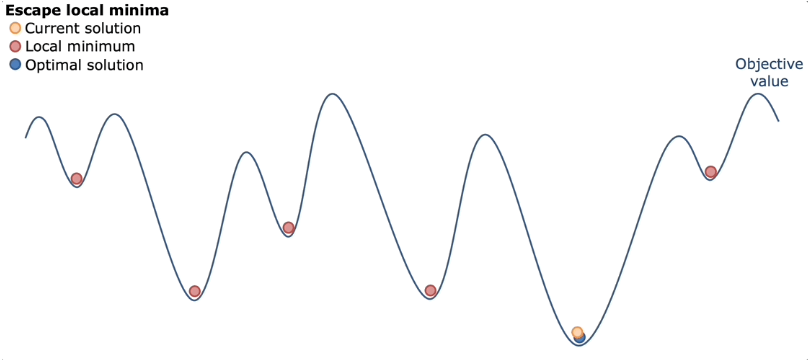
SA Numerical proof - Quod erat demonstratum



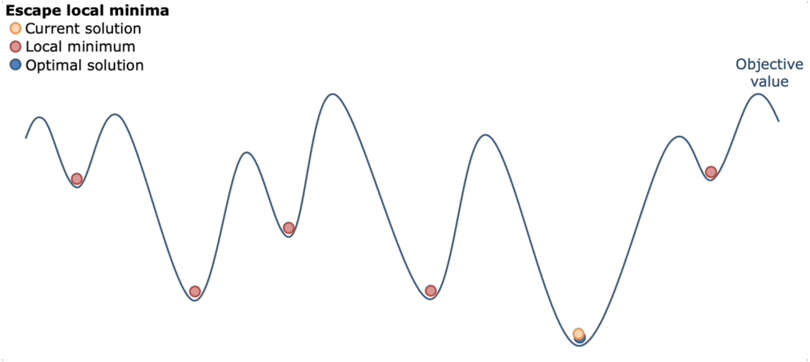
SA Numerical proof - Quod erat demonstratum



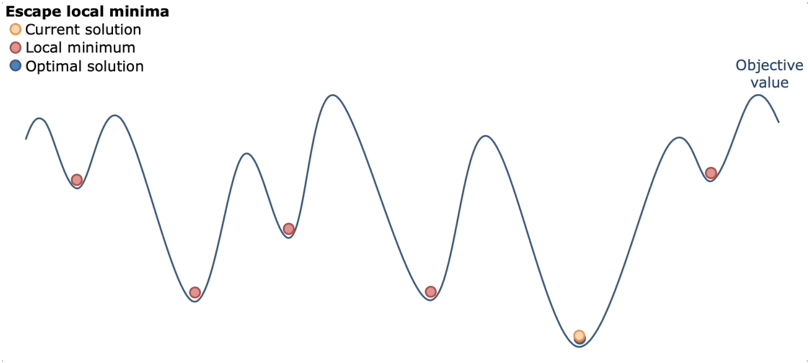
SA Numerical proof - Quod erat demonstratum



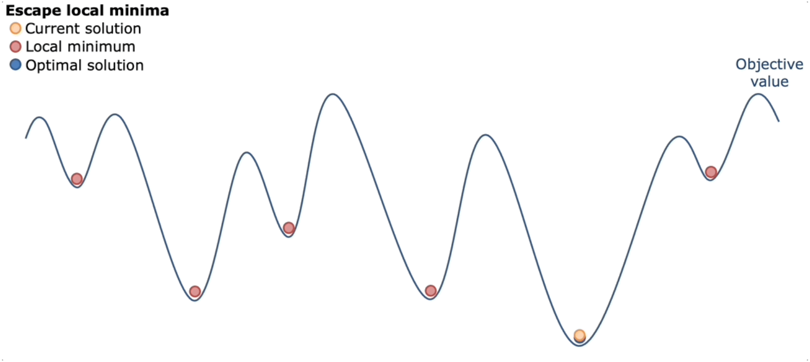
SA Numerical proof - Quod erat demonstratum



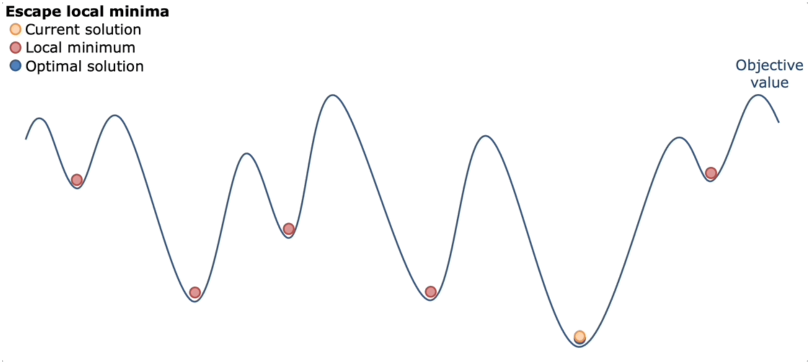
SA Numerical proof - Quod erat demonstratum



SA Numerical proof - Quod erat demonstratum



SA Numerical proof - Quod erat demonstratum



Option 2: Parallel/Interacting

$(X_0^i)_{1\leq i\leq N} \rightsquigarrow (X_1^i)_{1\leq i\leq N} \rightsquigarrow \ldots$ s.t.

$$\forall n \geq 0 \qquad (X_n^i)_{1\leq i\leq N} \quad \text{"almost" } N \text{ iid samples from } \quad \pi_{\mathbf{n}}$$

$$\Updownarrow$$

$$\forall n \geq 0 \qquad \frac{1}{\mathbf{N}} \sum_{1\leq i\leq \mathbf{N}} f(X_n^{\mathbf{i}}) \simeq_{\mathbf{N}\uparrow\infty} \int f(x)\, \pi_n(dx)$$

for **a given or a well chosen** **interpolating sequence** of target measures

$$\pi_0 \rightsquigarrow \pi_1 \rightsquigarrow \ldots \rightsquigarrow \pi_n \rightsquigarrow \ldots$$

Option 2: Parallel/Interacting

$(X_0^i)_{1\leq i\leq N} \rightsquigarrow (X_1^i)_{1\leq i\leq N} \rightsquigarrow \ldots$ s.t.

$$\forall n \geq 0 \qquad (X_n^i)_{1\leq i\leq N} \quad \text{"almost"} \; N \text{ iid samples from } \quad \pi_{\mathbf{n}}$$

$$\Updownarrow$$

$$\forall n \geq 0 \qquad \frac{1}{\mathbf{N}} \sum_{1\leq i\leq \mathbf{N}} f(X_n^{\mathbf{i}}) \;\simeq_{\mathbf{N}\uparrow\infty}\; \int f(x)\; \pi_n(dx)$$

for **a given or a well chosen** **interpolating sequence** of target measures

$$\pi_0 \rightsquigarrow \pi_1 \rightsquigarrow \ldots \rightsquigarrow \pi_n \rightsquigarrow \ldots$$

NOTE:

$\mathbf{N}$ = precision parameter = Computational power = Number of particles =...

Parallel version of SA $\rightsquigarrow$ Interpolating targets

Note that $\beta_n \uparrow \Leftrightarrow$ Interpolating targets π_{β_n}

$$\pi_{\beta_n}(dx) \;\propto\; e^{-(\beta_n - \beta_{n-1})U(x)}\; \pi_{\beta_{n-1}}(dx)$$

Parallel version of SA $\rightsquigarrow$ Interpolating targets

Note that $\beta_n \uparrow \Leftrightarrow$ Interpolating targets π_{β_n}

$$\pi_{\beta_n}(dx) \propto e^{-(\beta_n-\beta_{n-1})U(x)}\, \pi_{\beta_{n-1}}(dx)$$

Simple rule: Accept a sample "x" from $\pi_{\beta_{n-1}}$ *with probability* $e^{-(\beta_n-\beta_{n-1})U(x)}$

$$\Downarrow$$

$$\mathrm{Law}(\mathit{sample} \mid \mathit{acceptance}) = \pi_{\beta_n}$$

Parallel version of SA $\rightsquigarrow$ Interpolating targets

Note that $\beta_n \uparrow \Leftrightarrow$ Interpolating targets π_{β_n}

$$\pi_{\beta_n}(dx) \ \propto \ e^{-(\beta_n-\beta_{n-1})U(x)} \ \pi_{\beta_{n-1}}(dx)$$

Simple rule: Accept a sample "x" from $\pi_{\beta_{n-1}}$ *with probability* $e^{-(\beta_n-\beta_{n-1})U(x)}$

$$\Downarrow$$

$$\mathrm{Law}(\textit{sample} \mid \textit{acceptance}) = \pi_{\beta_n}$$

Normalizing constants $\rightsquigarrow$ product formula

$$\mathcal{Z}_{\beta_n}/\mathcal{Z}_{\beta_{n-1}} = \nu(e^{-\beta_n U})/\nu(e^{-\beta_{n-1} U}) = \ \pi_{\beta_{n-1}}(e^{-(\beta_n-\beta_{n-1})U})$$

Interpolating targets $\rightsquigarrow$ Genetic Algorithms = **GA**

N "interacting walkers/individuals/particles/genes/..."

$$\xi_n := \begin{pmatrix} \xi_n^1 \\ \vdots \\ \xi_n^N \end{pmatrix} \quad \text{"almost" } N \text{ iid} \sim \pi_{\beta_n}$$

Interpolating targets $\rightsquigarrow$ Genetic Algorithms = **GA**

***N* "interacting walkers/individuals/particles/genes/..."**

$$\xi_n := \begin{pmatrix} \xi_n^1 \\ \vdots \\ \xi_n^N \end{pmatrix} \quad \text{"almost" } \mathbf{N} \text{ iid} \sim \pi_{\beta_n} \Longleftrightarrow \pi_{\beta_n}^{\mathbf{N}} := \frac{1}{N} \sum_{1 \le i \le N} \delta_{\xi_n^i} \simeq_{\mathbf{N} \uparrow \infty} \pi_{\beta_n}$$

Interpolating targets $\rightsquigarrow$ Genetic Algorithms $=$ **GA**

N **"interacting walkers/individuals/particles/genes/..."**

$$\xi_n := \left(\begin{array}{c} \xi_n^1 \\ \vdots \\ \xi_n^N \end{array}\right) \quad \text{"almost" N iid} \sim \pi_{\beta_n} \Longleftrightarrow \pi_{\beta_n}^{\mathbf{N}} := \frac{1}{N}\sum_{1\leq i\leq N} \delta_{\xi_n^i} \simeq_{\mathbf{N}\uparrow\infty} \pi_{\beta_n}$$

- **Selection:** Accept/Select the individuals adapted to the next temperature parameter β_{n+1} using the weights fitness functions:

$$x \mapsto \exp\left(-(\beta_{n+1}-\beta_n)U(x)\right)$$

Interpolating targets $\rightsquigarrow$ Genetic Algorithms = **GA**

***N* "interacting walkers/individuals/particles/genes/..."**

$$\xi_n := \begin{pmatrix} \xi_n^1 \\ \vdots \\ \xi_n^N \end{pmatrix} \quad \text{"almost" N iid} \sim \pi_{\beta_n} \Longleftrightarrow \pi_{\beta_n}^{\mathbf{N}} := \frac{1}{N} \sum_{1 \le i \le N} \delta_{\xi_n^i} \simeq_{\mathbf{N}\uparrow\infty} \pi_{\beta_n}$$

- **Selection:** Accept/Select the individuals adapted to the next temperature parameter β_{n+1} using the weights fitness functions:

$$x \mapsto \exp\left(-(\beta_{n+1} - \beta_n) U(x)\right)$$

Key observation: The selected individuals, say $\widehat{\xi}_n = (\widehat{\xi}_n^i)_{1 \le i \le N}$ are adapted to $\pi_{\beta_{n+1}}$.

$$\widehat{\xi}_n := \begin{pmatrix} \widehat{\xi}_n^1 \\ \vdots \\ \widehat{\xi}_n^N \end{pmatrix} \quad \textbf{"almost" } N \textbf{ iid} \sim \pi_{\beta_{n+1}}$$

Interpolating targets $\rightsquigarrow$ Genetic Algorithms = **GA**

***N* "interacting walkers/individuals/particles/genes/..."**

$$\xi_n := \begin{pmatrix} \xi_n^1 \\ \vdots \\ \xi_n^N \end{pmatrix} \quad \text{"almost" N iid} \sim \pi_{\beta_n} \Longleftrightarrow \pi_{\beta_n}^{\mathbf{N}} := \frac{1}{N} \sum_{1 \leq i \leq N} \delta_{\xi_n^i} \simeq_{\mathbf{N} \uparrow \infty} \pi_{\beta_n}$$

- **Selection:** Accept/Select the individuals adapted to the next temperature parameter β_{n+1} using the weights fitness functions:

$$x \mapsto \exp\left(-(\beta_{n+1} - \beta_n) U(x)\right)$$

Key observation: The selected individuals, say $\widehat{\xi}_n = (\widehat{\xi}_n^i)_{1 \leq i \leq N}$ are adapted to $\pi_{\beta_{n+1}}$.

$$\widehat{\xi}_n := \begin{pmatrix} \widehat{\xi}_n^1 \\ \vdots \\ \widehat{\xi}_n^N \end{pmatrix} \quad \text{"almost" } N \text{ iid} \sim \pi_{\beta_{n+1}}$$

- **Mutation:** $\widehat{\xi}_n \rightsquigarrow \xi_{n+1}$ with a $\pi_{\beta_{n+1}}$-shaker (**one/twice/...**).

Normalizing constants (unbiased)

$$\frac{\mathcal{Z}_{\beta_n}}{\mathcal{Z}_{\beta_0}} \simeq_{N\uparrow\infty} \prod_{0\leq k<n} \frac{1}{N} \sum_{1\leq i\leq N} \exp\left(-(\beta_{k+1}-\beta_k)U\left(i\text{-th } \pi_{\beta_k}\text{-shakers}\right)\right)$$

Normalizing constants (unbiased)

$$\frac{\mathcal{Z}_{\beta_n}}{\mathcal{Z}_{\beta_0}} \simeq_{N\uparrow\infty} \prod_{0\leq k<n} \frac{1}{N} \sum_{1\leq i\leq N} \exp\left(-(\beta_{k+1}-\beta_k)U\left(i\text{-th } \pi_{\beta_k}\text{-shakers}\right)\right)$$

Adaptive version:

Choose sequentially $\beta_{k+1} \geq \beta_k$ s.t. the weighted average

$$\beta_{k+1} \in [\beta_k, \infty[\mapsto \frac{1}{N} \sum_{1\leq i\leq N} \exp\left(-(\beta_{k+1}-\beta_k)U\left(i\text{-th } \pi_{\beta_k}\text{-shakers}\right)\right)$$

is above some threshold

Normalizing constants (unbiased)

$$\frac{\mathcal{Z}_{\beta_n}}{\mathcal{Z}_{\beta_0}} \simeq_{N\uparrow\infty} \prod_{0\leq k<n} \frac{1}{N} \sum_{1\leq i\leq N} \exp\left(-(\beta_{k+1}-\beta_k) U\left(i\text{-th } \pi_{\beta_k}\text{-shakers}\right)\right)$$

Adaptive version:

Choose sequentially $\beta_{k+1} \geq \beta_k$ s.t. the weighted average

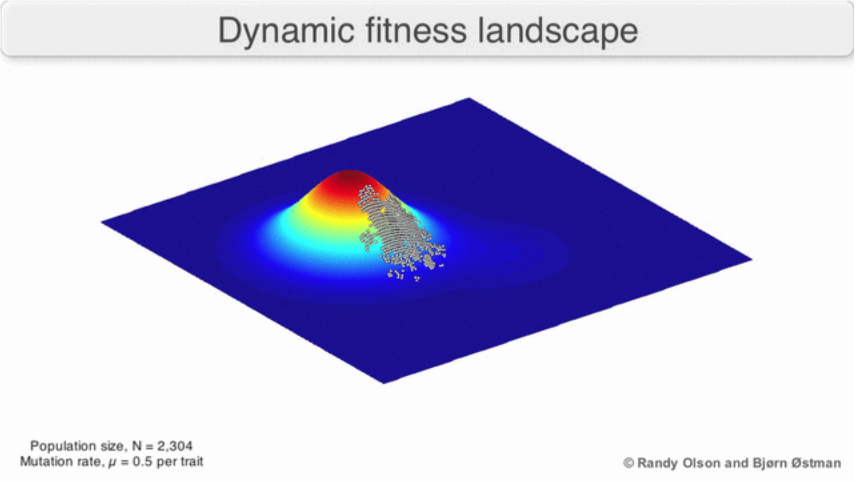
$$\beta_{k+1} \in [\beta_k, \infty[\mapsto \frac{1}{N} \sum_{1\leq i\leq N} \exp\left(-(\beta_{k+1}-\beta_k) U\left(i\text{-th } \pi_{\beta_k}\text{-shakers}\right)\right)$$

is above some threshold

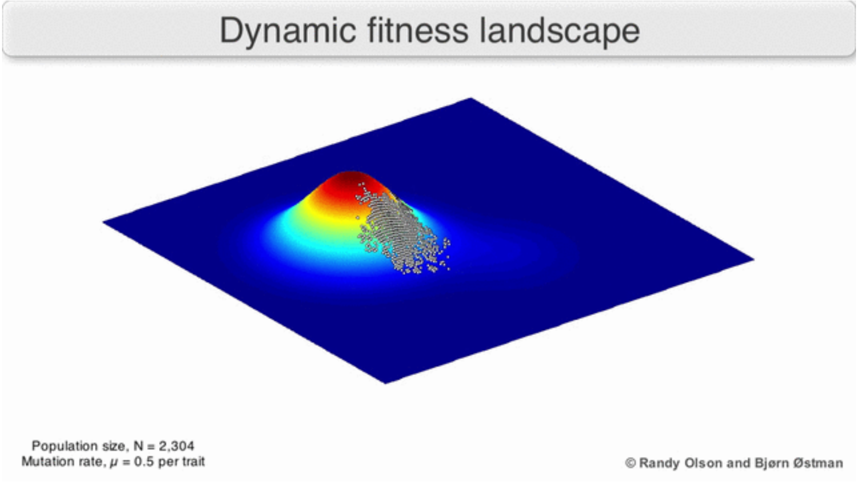
Potential variations ($\uparrow$ dimensions, modes,...):

$$U_k \rightsquigarrow U_{k+1}$$

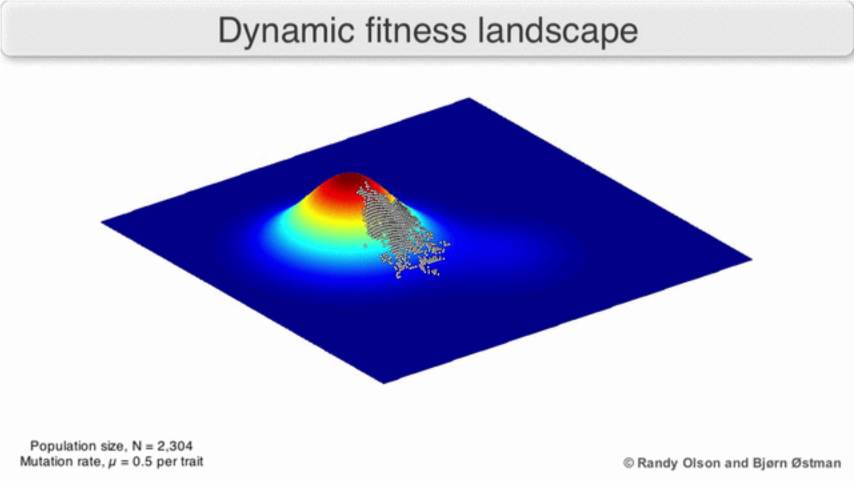
GA Numerical proof - Quod erat demonstratum



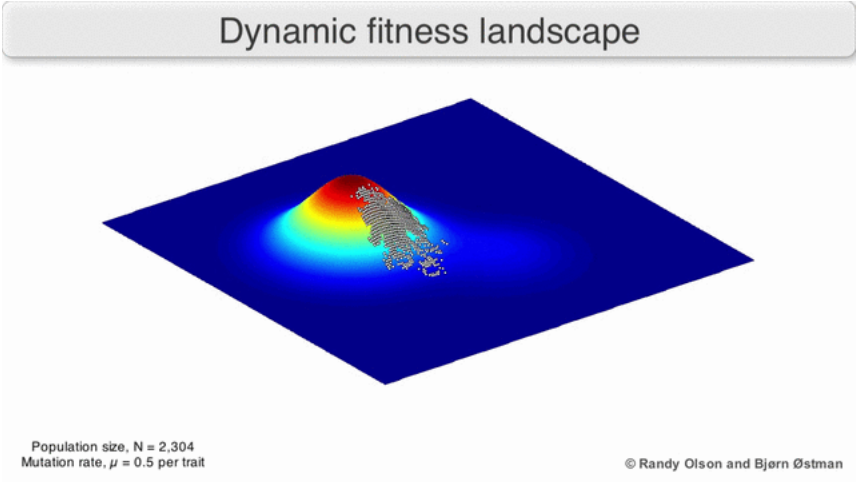
GA Numerical proof - Quod erat demonstratum



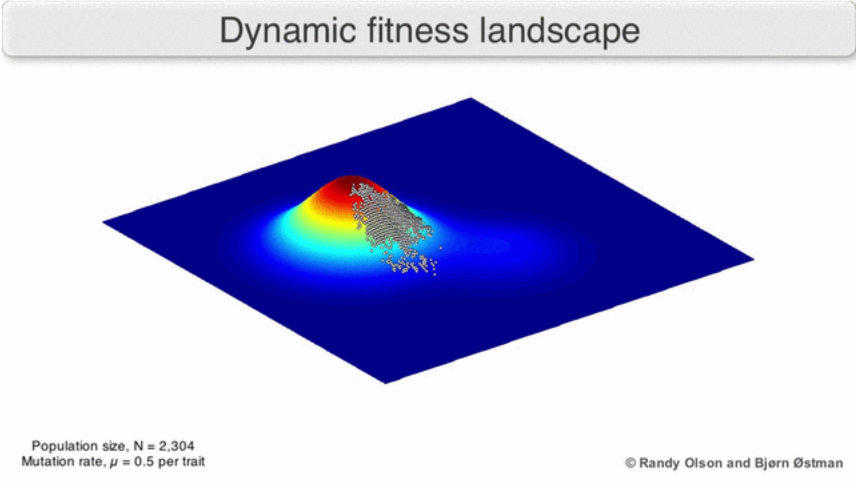
GA Numerical proof - Quod erat demonstratum



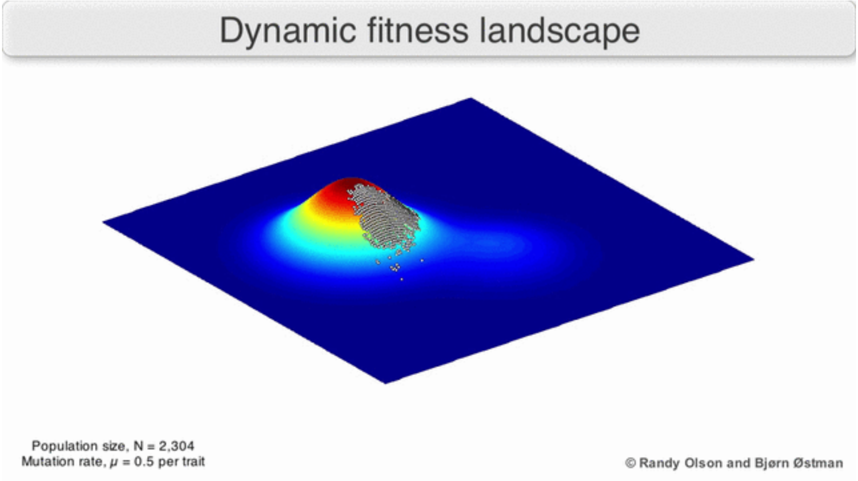
GA Numerical proof - Quod erat demonstratum



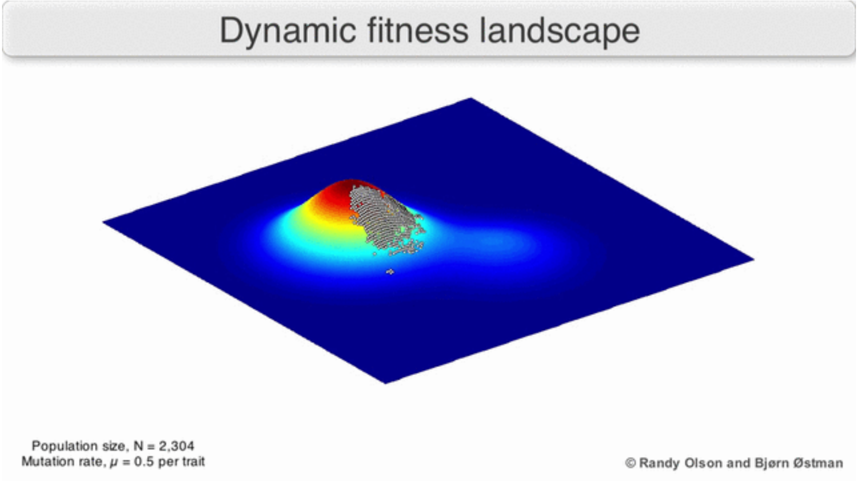
GA Numerical proof - Quod erat demonstratum



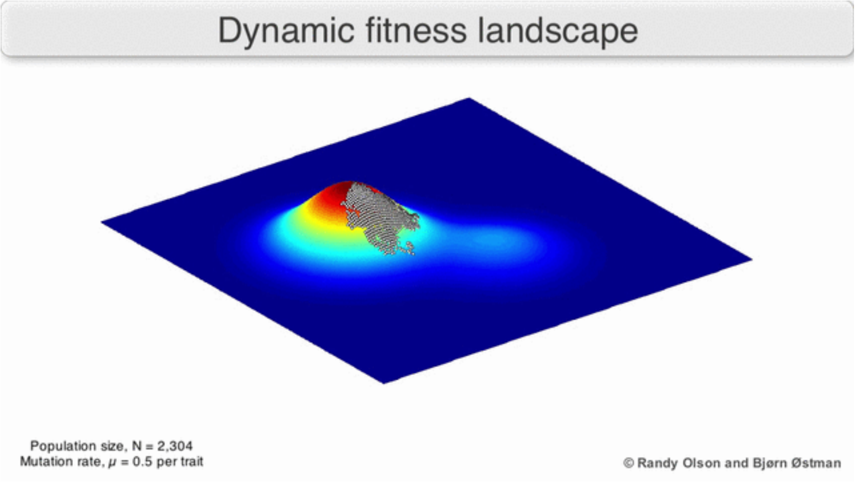
GA Numerical proof - Quod erat demonstratum



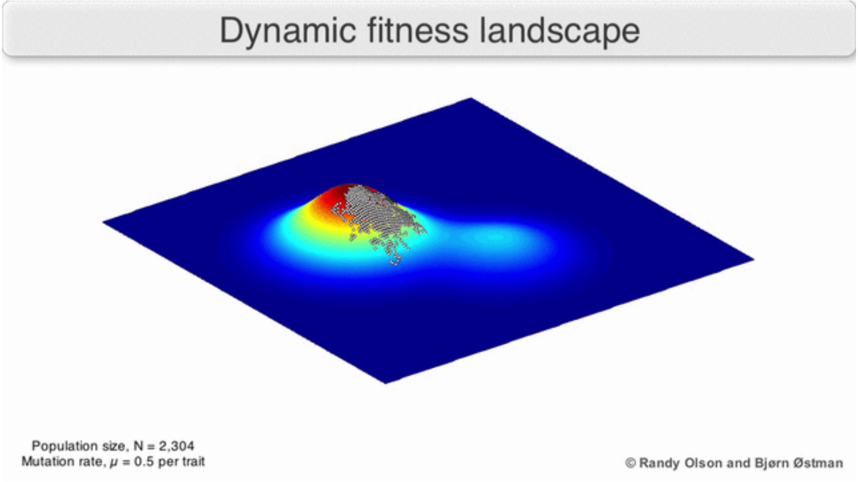
GA Numerical proof - Quod erat demonstratum



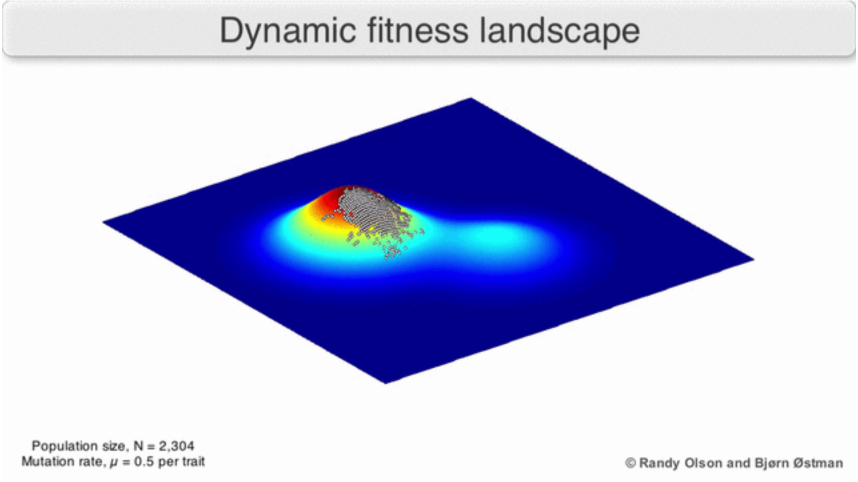
GA Numerical proof - Quod erat demonstratum



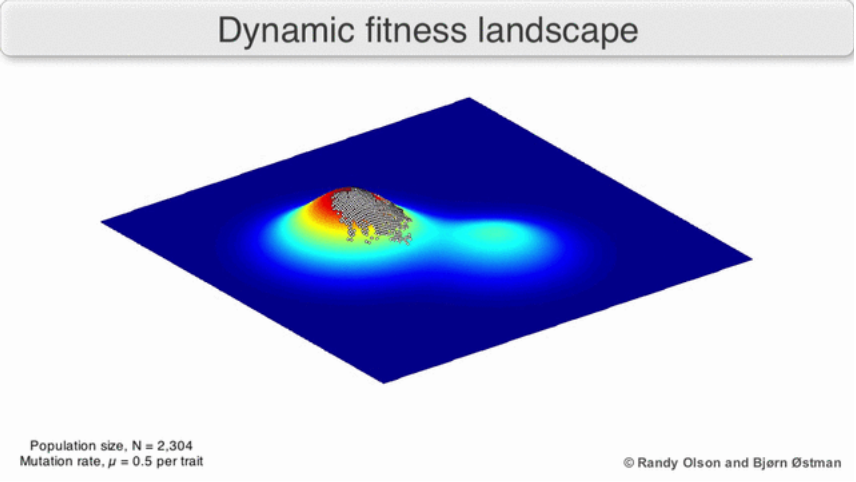
GA Numerical proof - Quod erat demonstratum



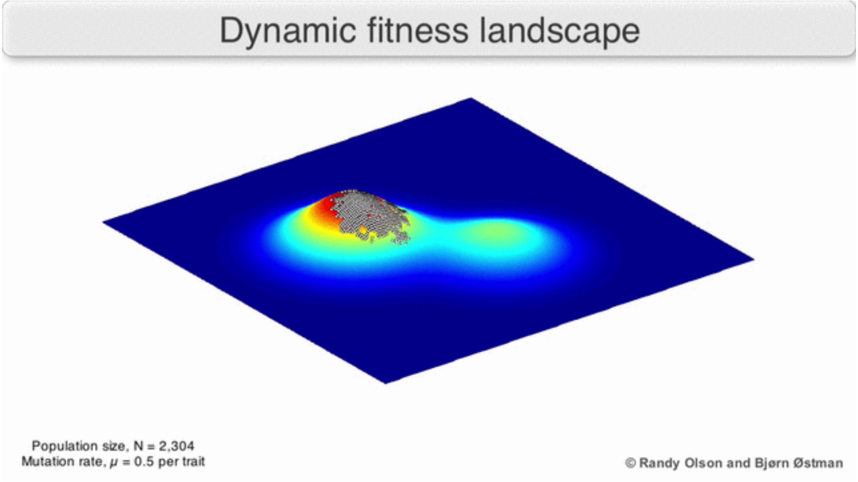
GA Numerical proof - Quod erat demonstratum



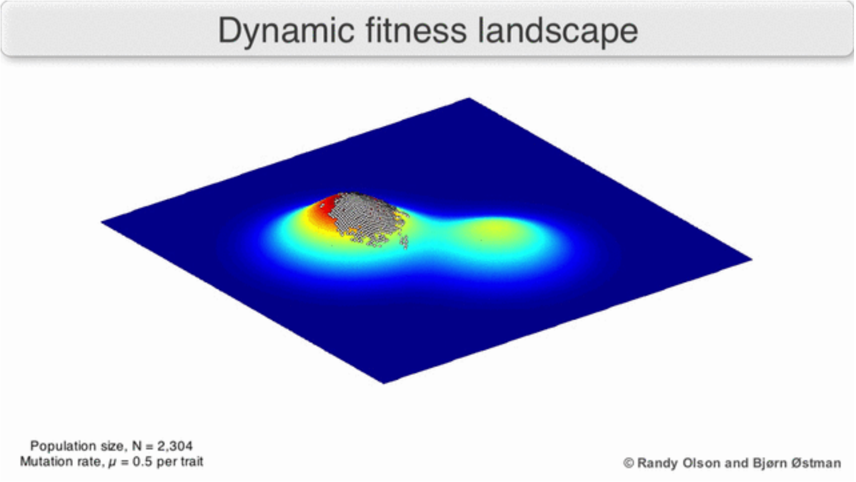
GA Numerical proof - Quod erat demonstratum



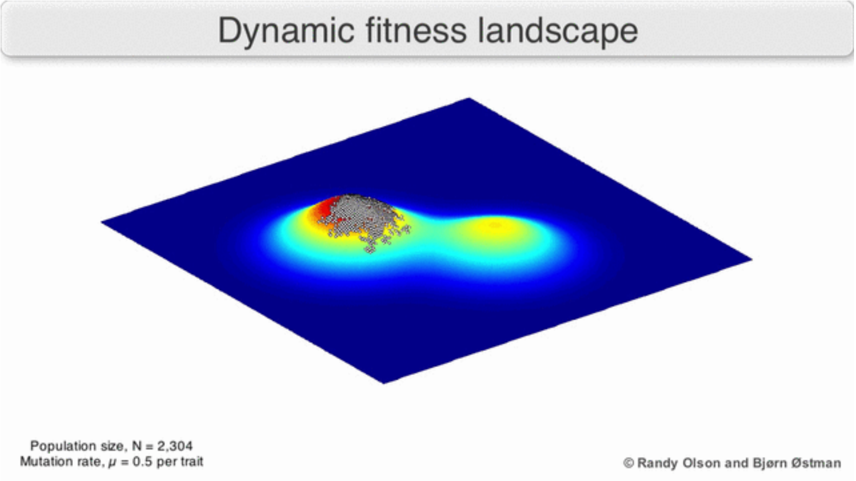
GA Numerical proof - Quod erat demonstratum



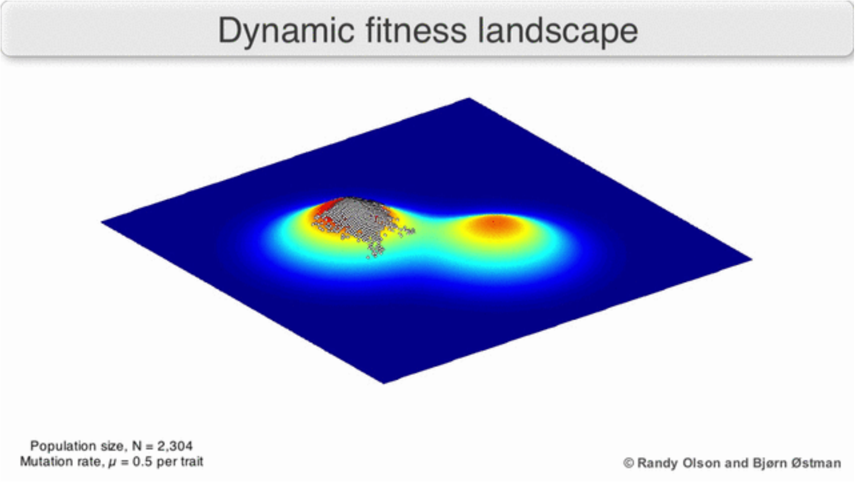
GA Numerical proof - Quod erat demonstratum



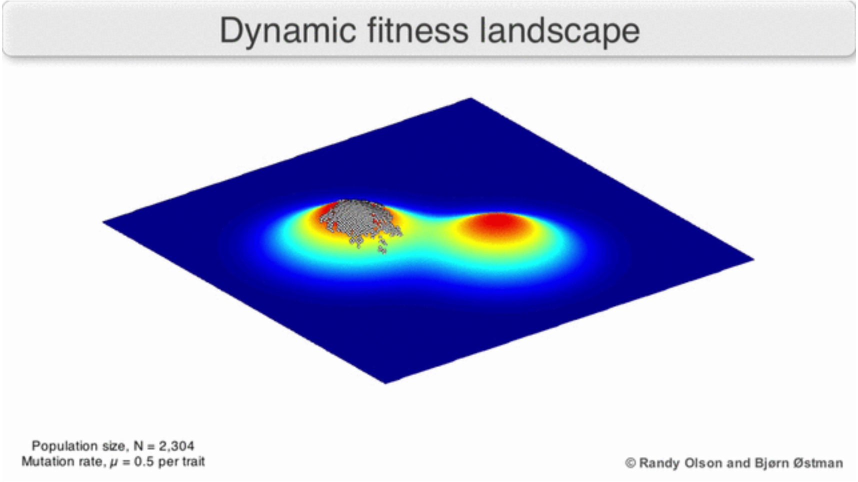
GA Numerical proof - Quod erat demonstratum



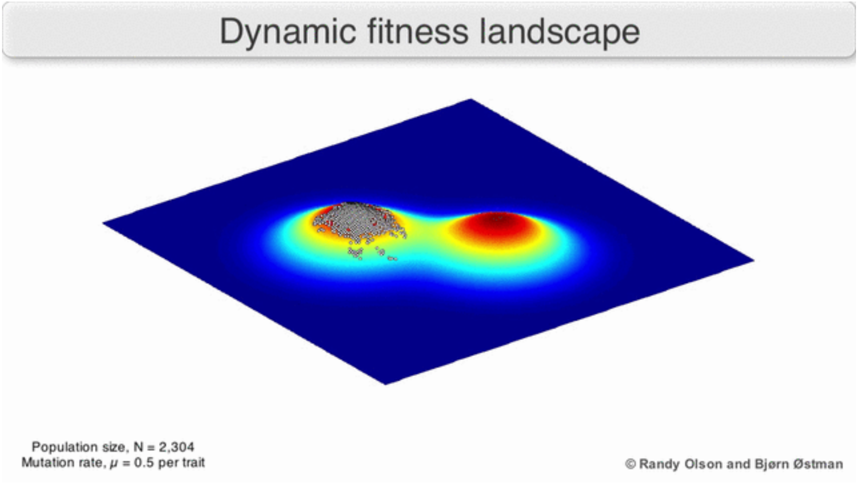
GA Numerical proof - Quod erat demonstratum



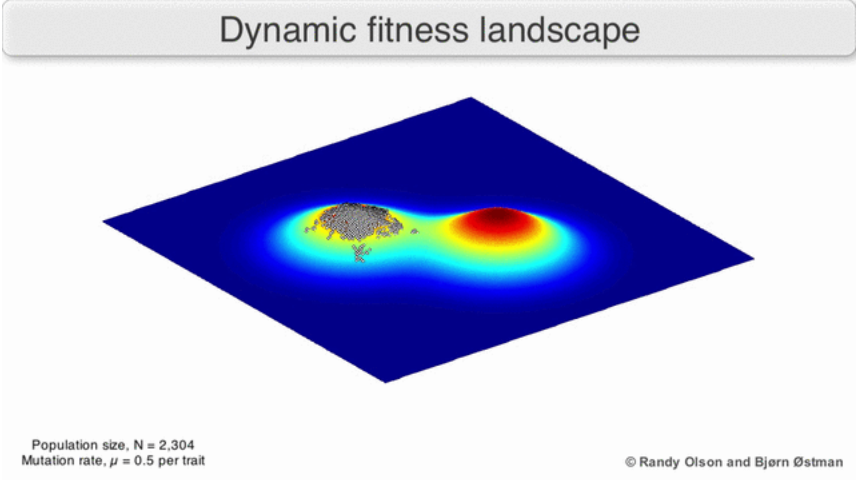
GA Numerical proof - Quod erat demonstratum



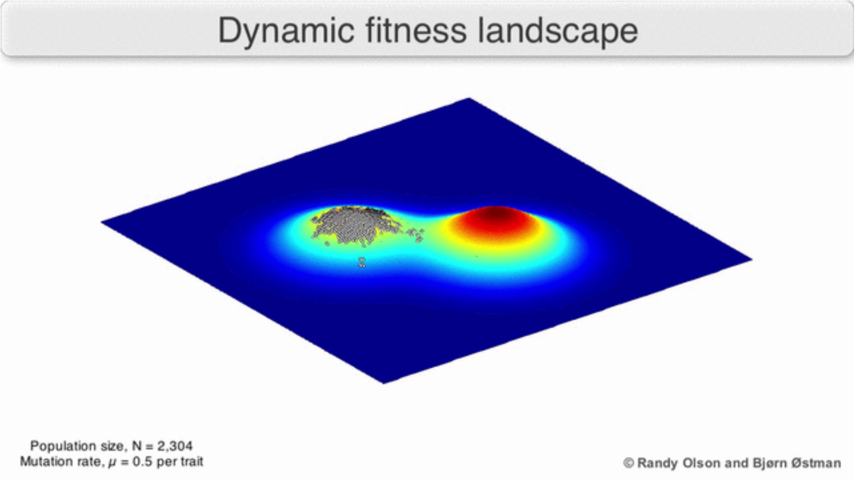
GA Numerical proof - Quod erat demonstratum



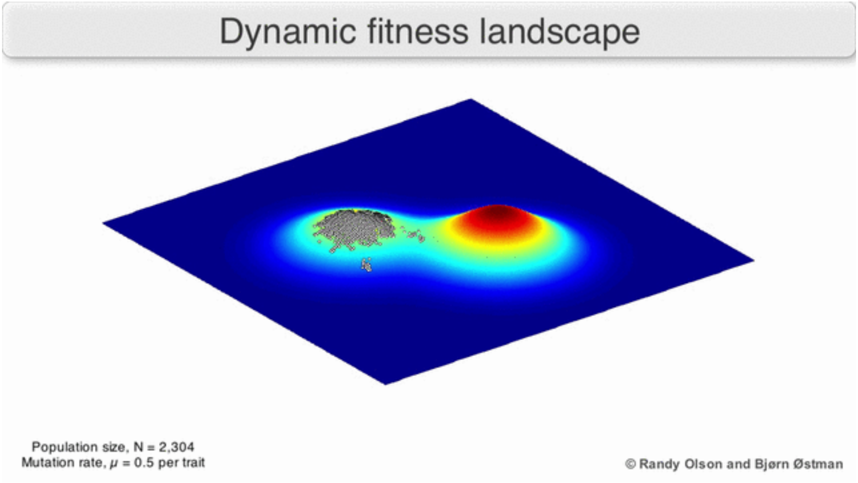
GA Numerical proof - Quod erat demonstratum



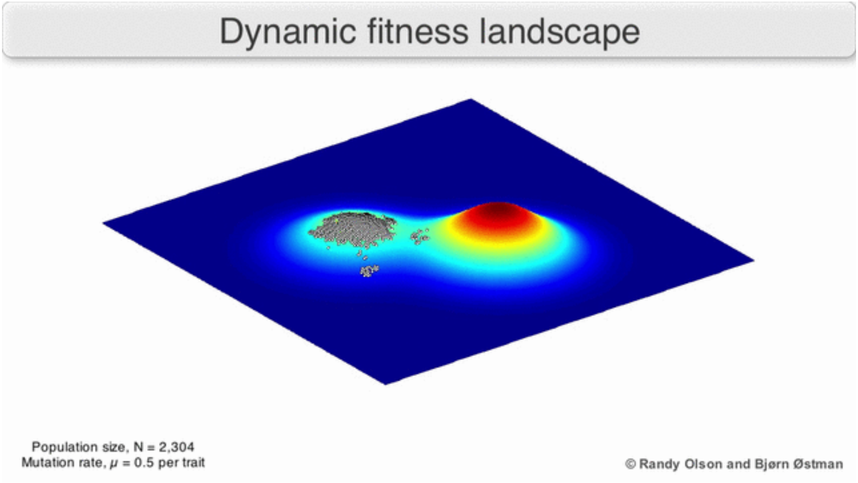
GA Numerical proof - Quod erat demonstratum



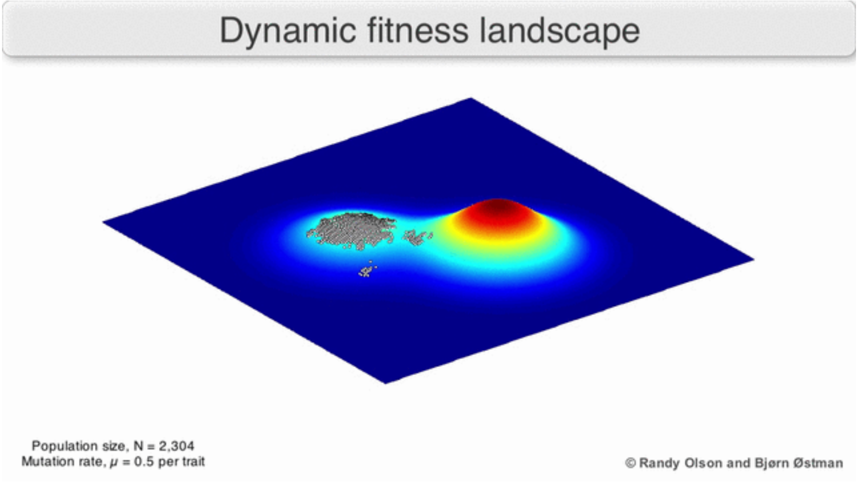
GA Numerical proof - Quod erat demonstratum



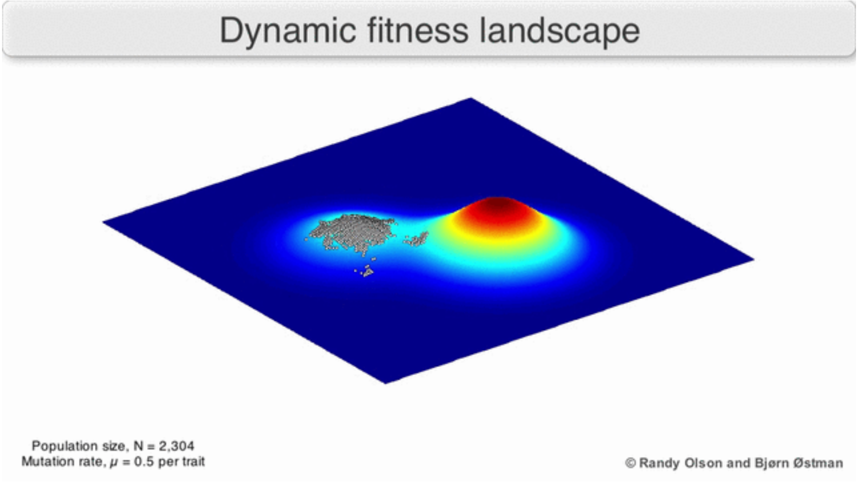
GA Numerical proof - Quod erat demonstratum



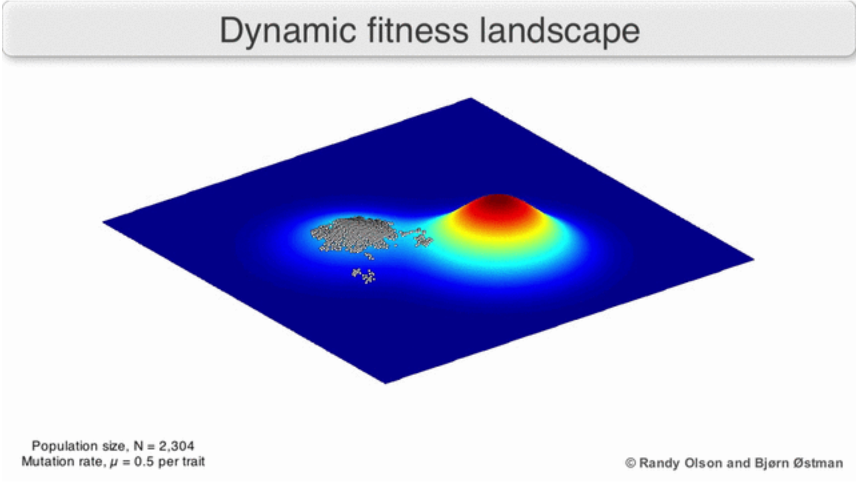
GA Numerical proof - Quod erat demonstratum



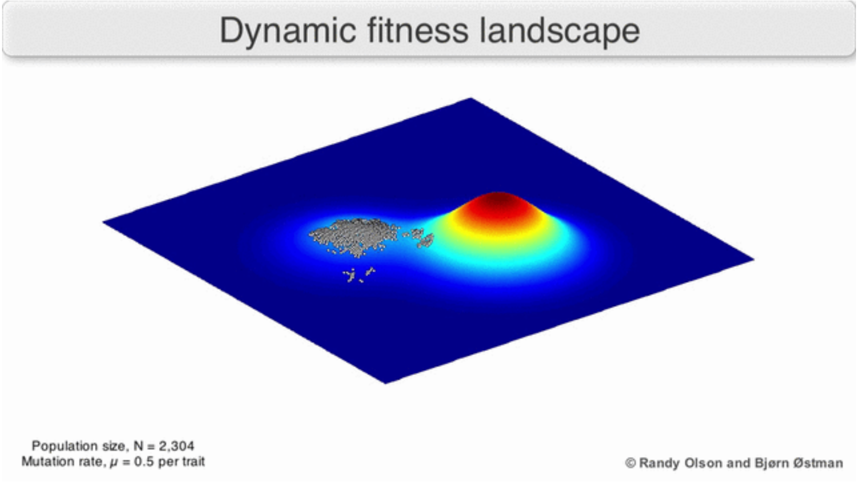
GA Numerical proof - Quod erat demonstratum



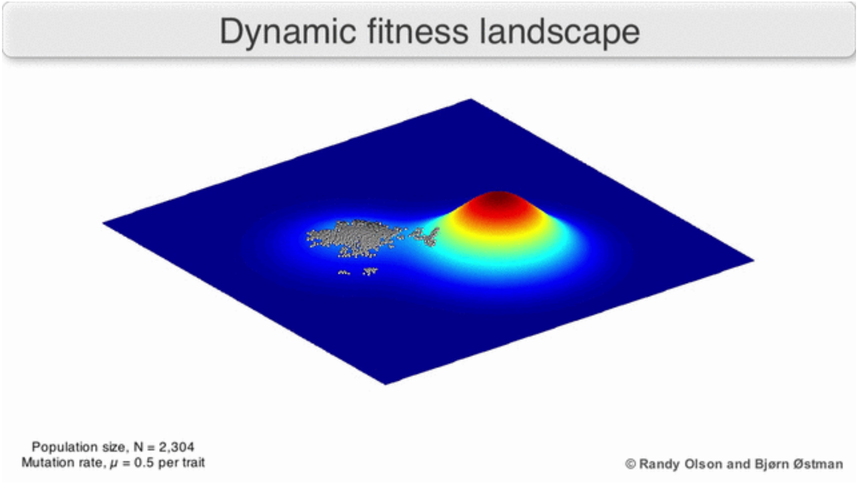
GA Numerical proof - Quod erat demonstratum



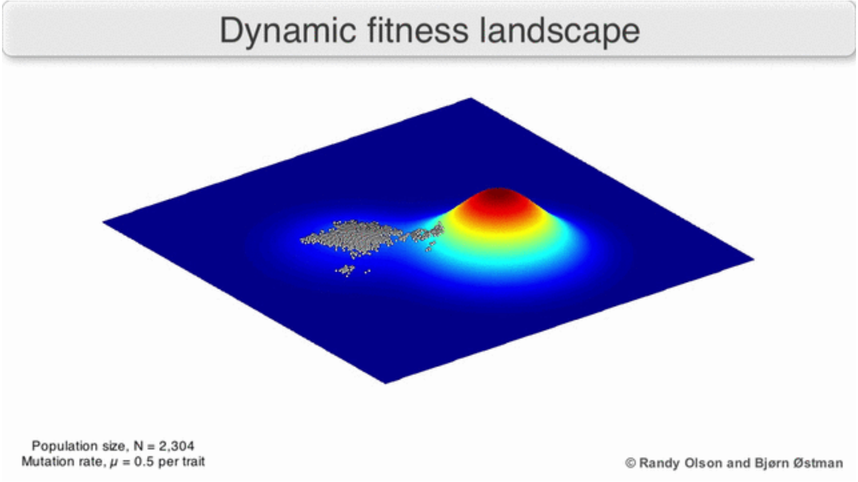
GA Numerical proof - Quod erat demonstratum



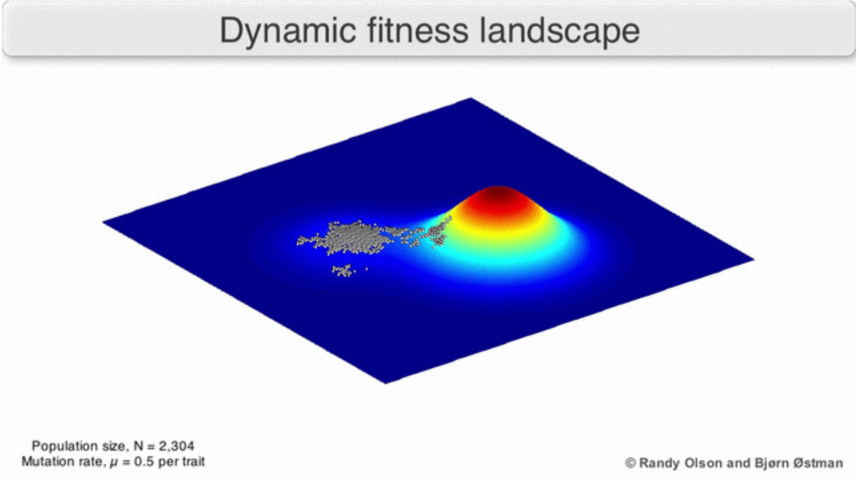
GA Numerical proof - Quod erat demonstratum



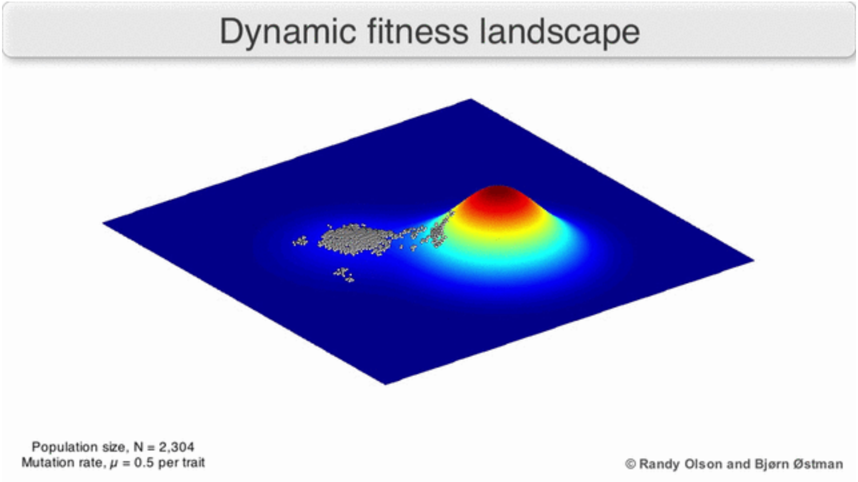
GA Numerical proof - Quod erat demonstratum



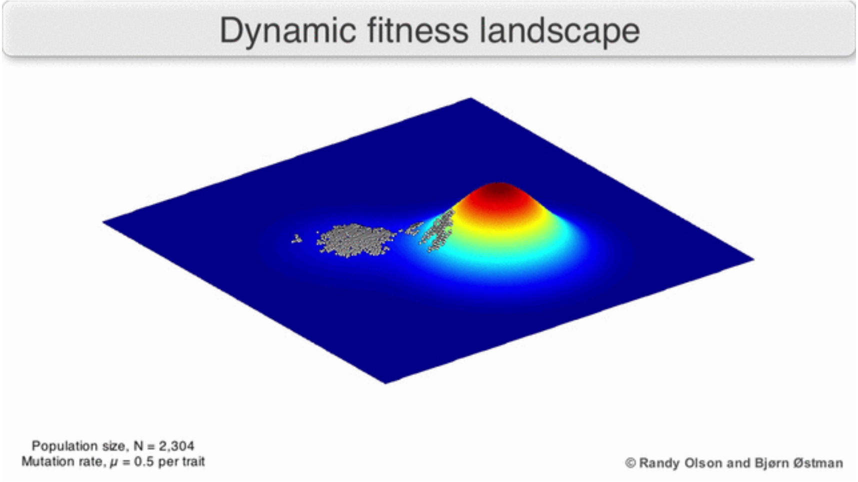
GA Numerical proof - Quod erat demonstratum



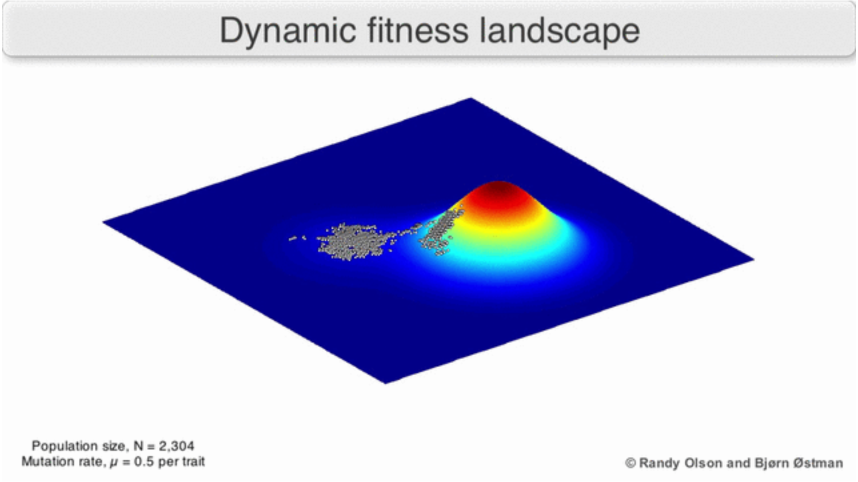
GA Numerical proof - Quod erat demonstratum



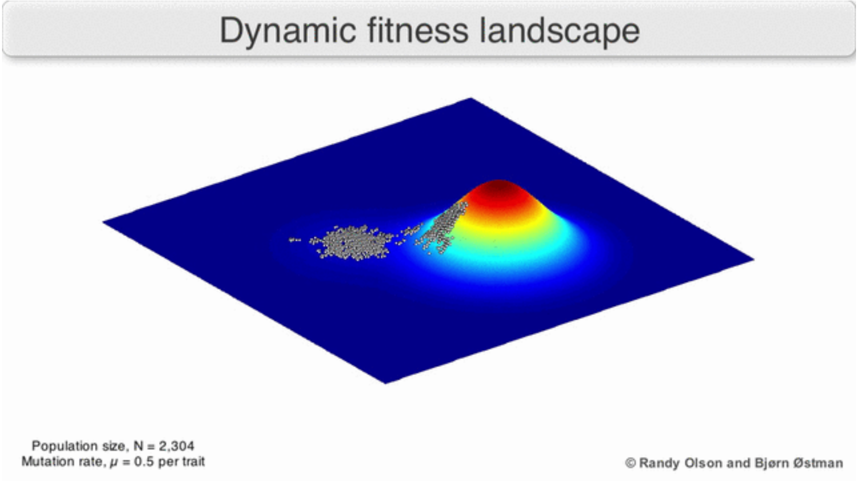
GA Numerical proof - Quod erat demonstratum



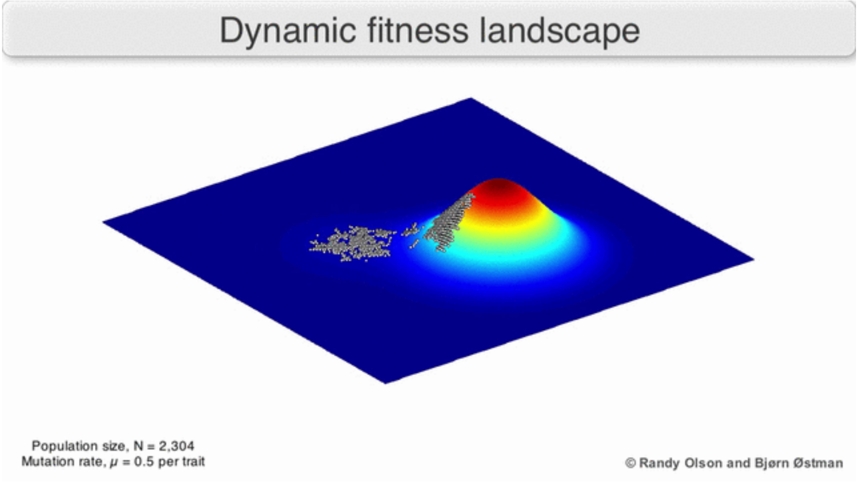
GA Numerical proof - Quod erat demonstratum



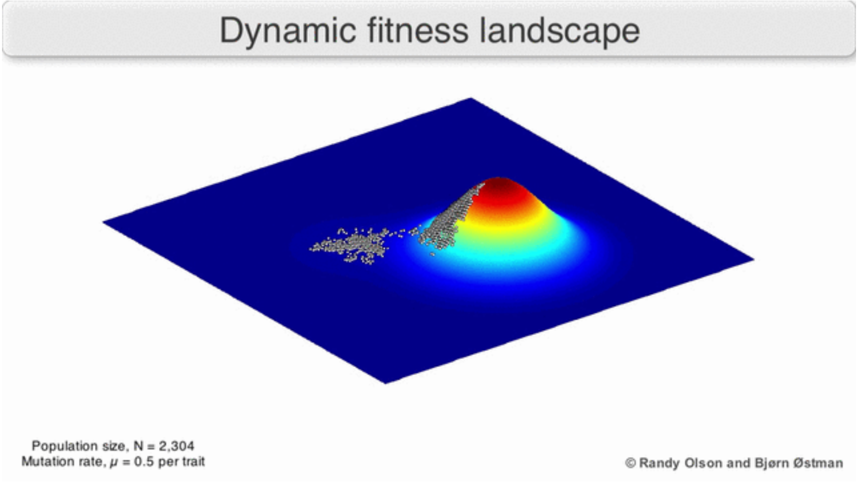
GA Numerical proof - Quod erat demonstratum



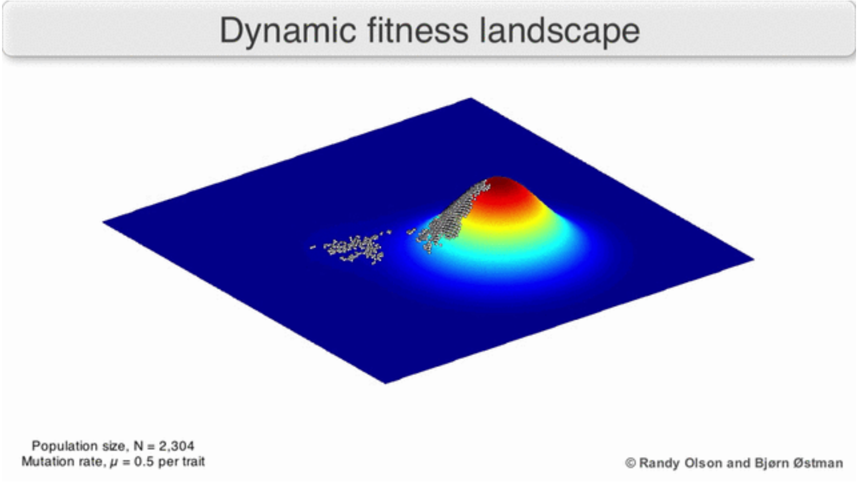
GA Numerical proof - Quod erat demonstratum



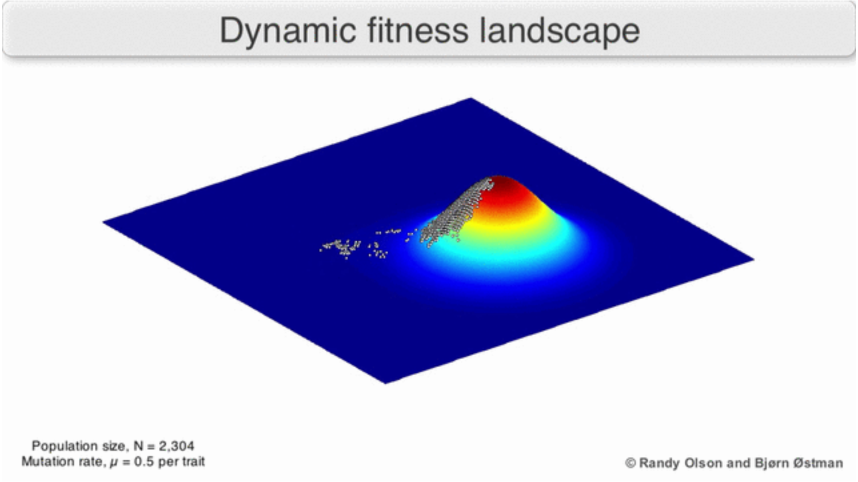
GA Numerical proof - Quod erat demonstratum



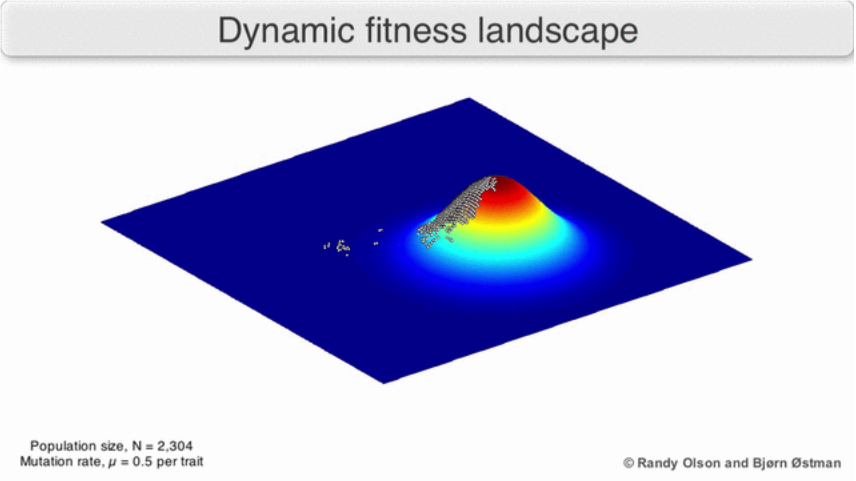
GA Numerical proof - Quod erat demonstratum



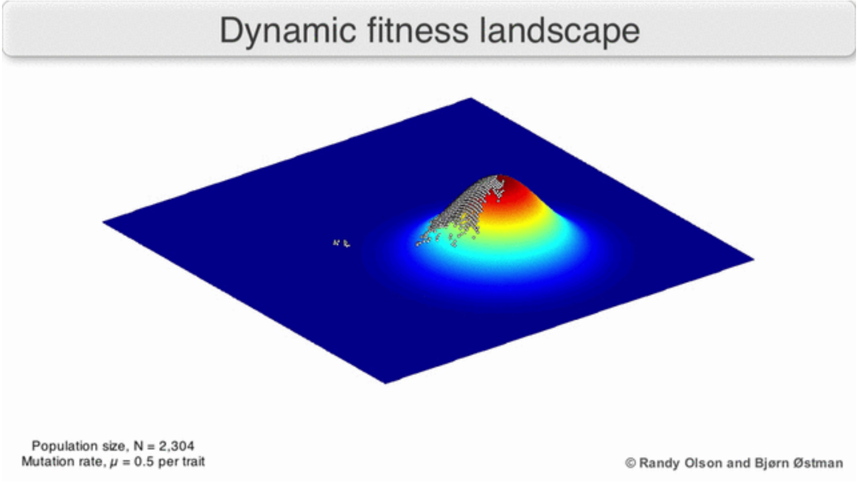
GA Numerical proof - Quod erat demonstratum



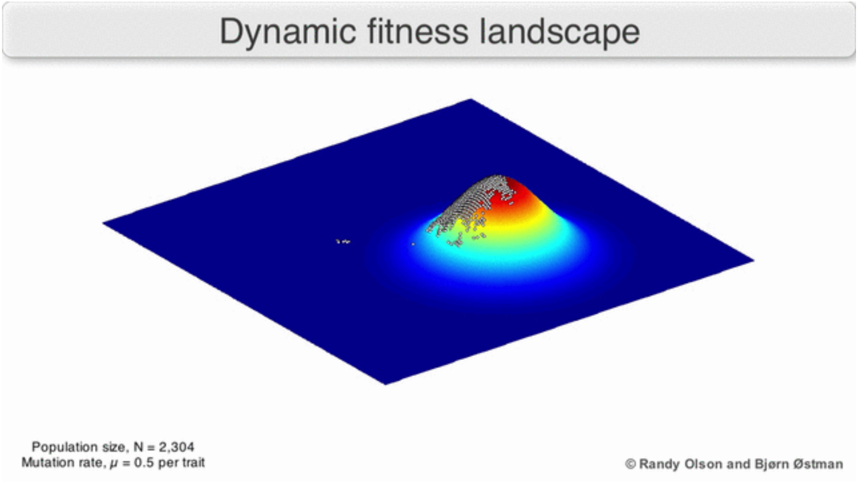
GA Numerical proof - Quod erat demonstratum



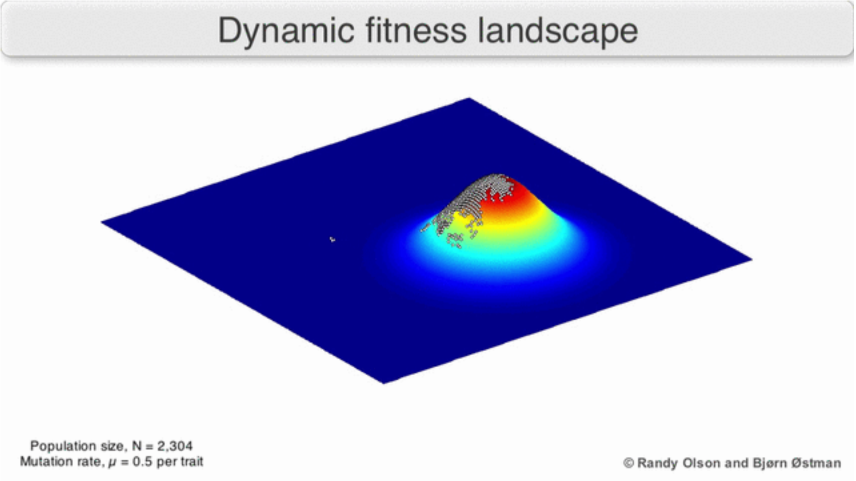
GA Numerical proof - Quod erat demonstratum



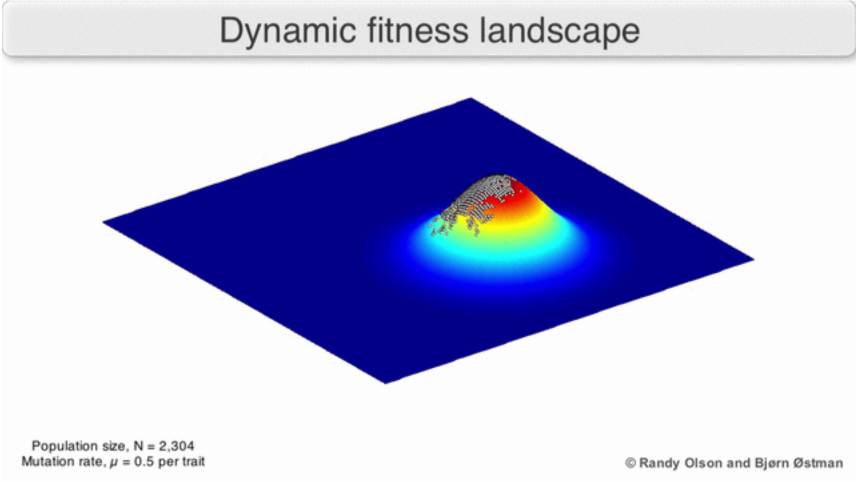
GA Numerical proof - Quod erat demonstratum



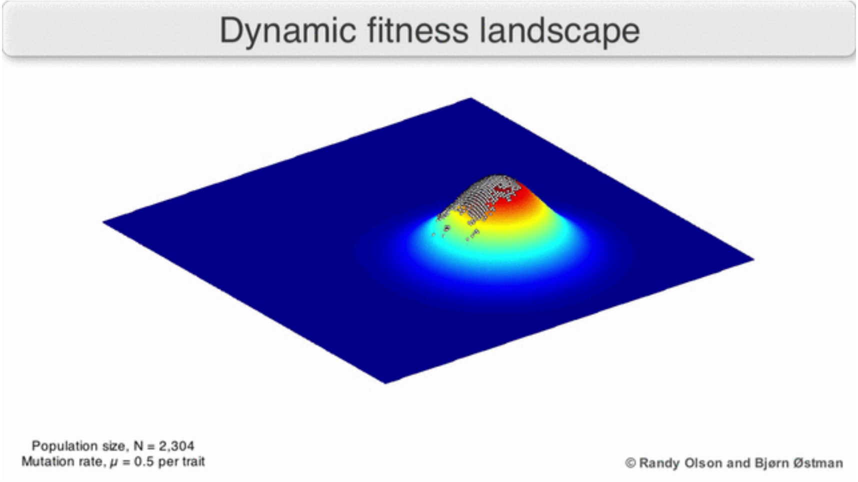
GA Numerical proof - Quod erat demonstratum



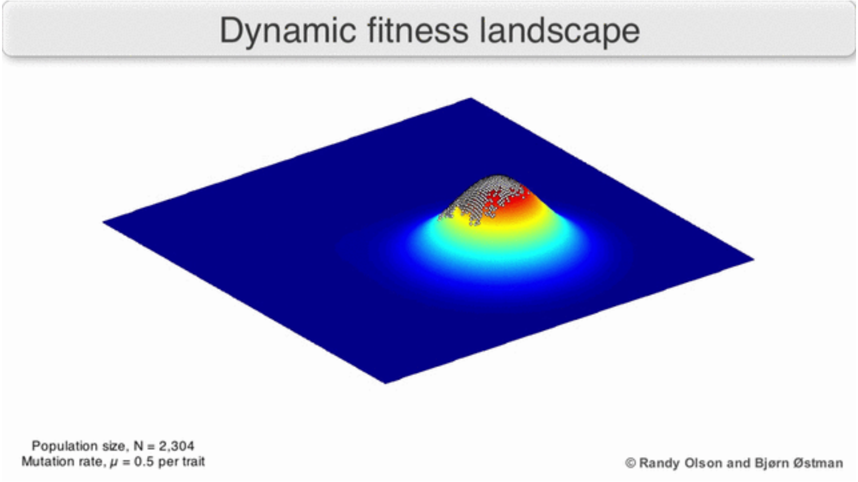
GA Numerical proof - Quod erat demonstratum



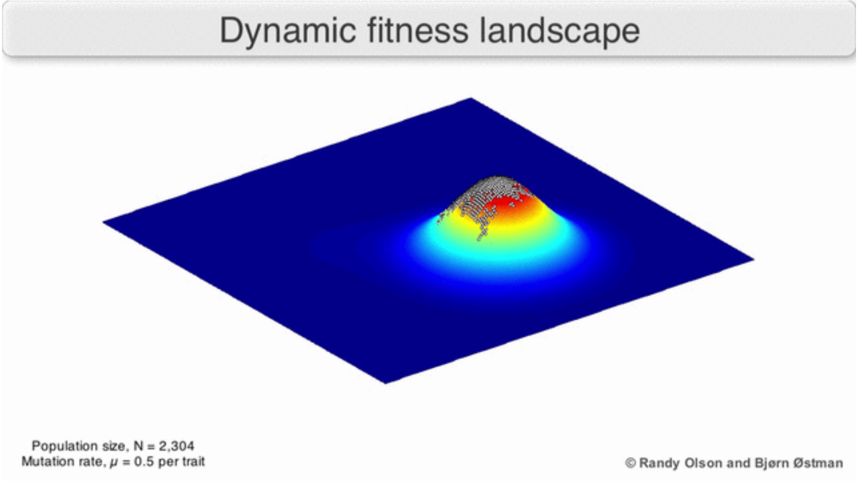
GA Numerical proof - Quod erat demonstratum



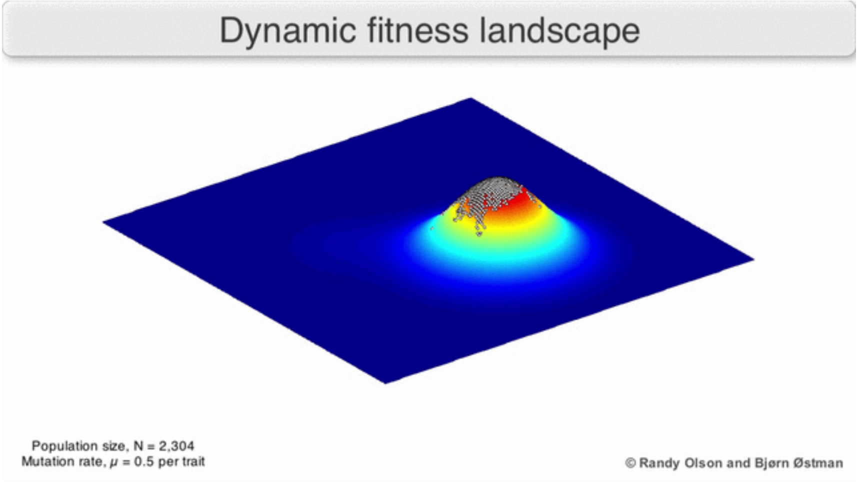
GA Numerical proof - Quod erat demonstratum



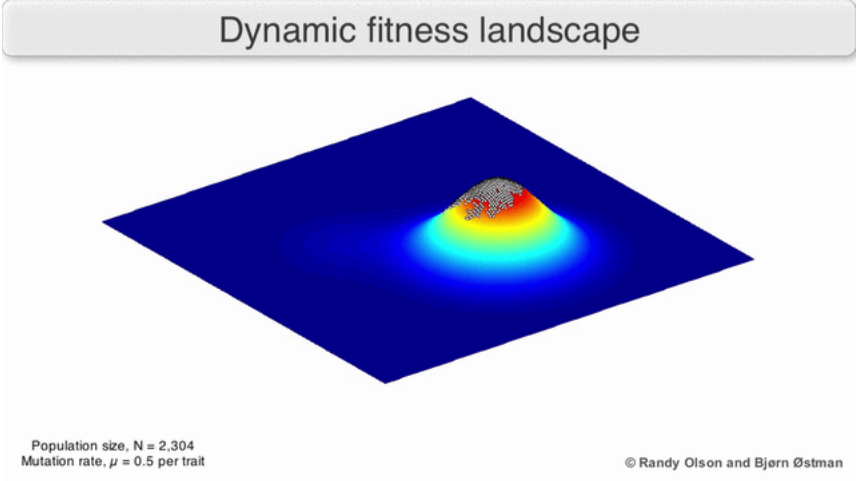
GA Numerical proof - Quod erat demonstratum



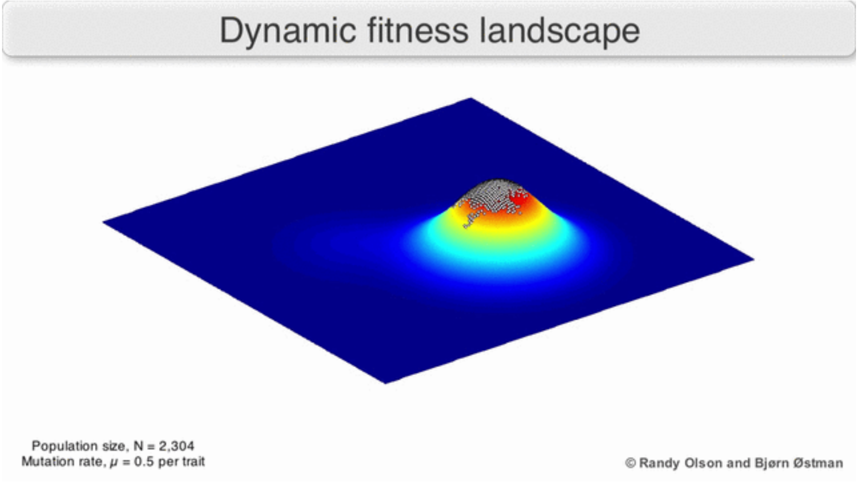
GA Numerical proof - Quod erat demonstratum



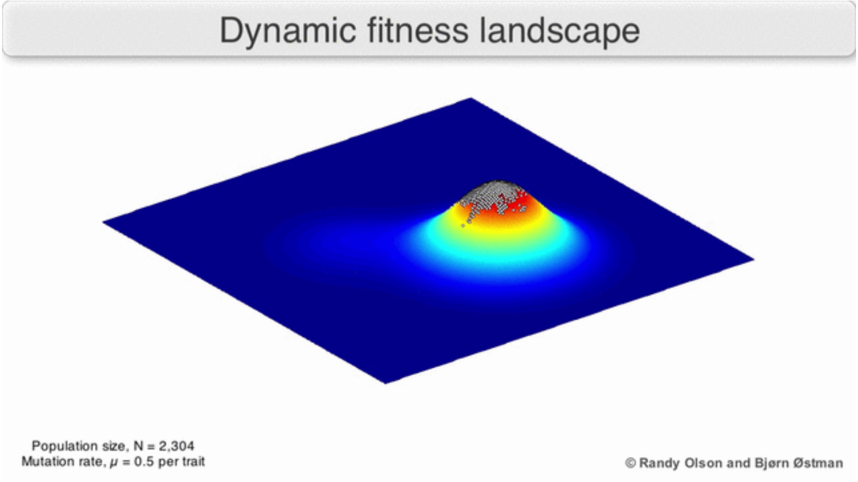
GA Numerical proof - Quod erat demonstratum



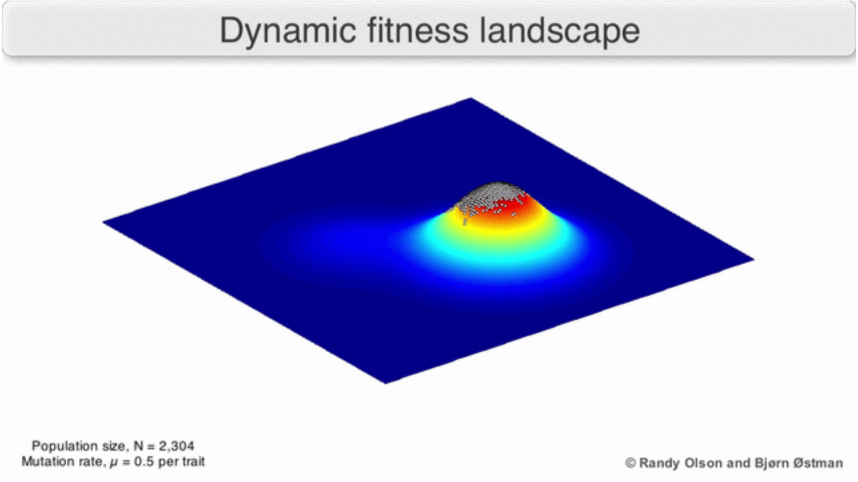
GA Numerical proof - Quod erat demonstratum



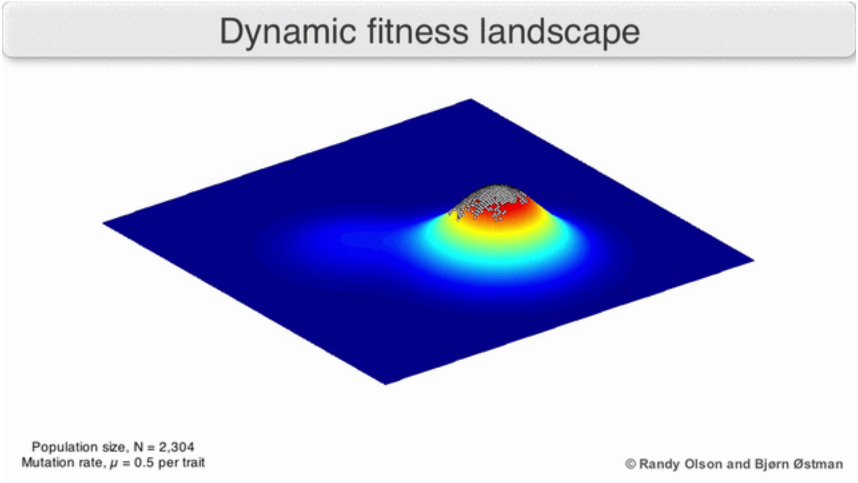
GA Numerical proof - Quod erat demonstratum



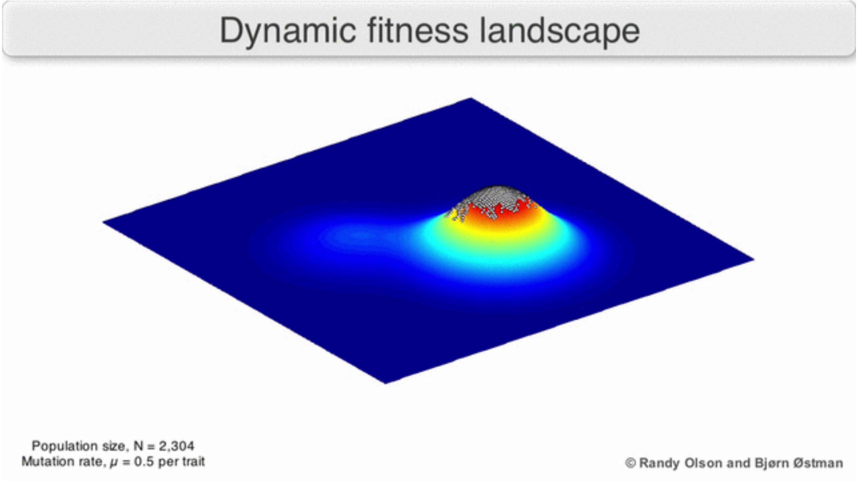
GA Numerical proof - Quod erat demonstratum



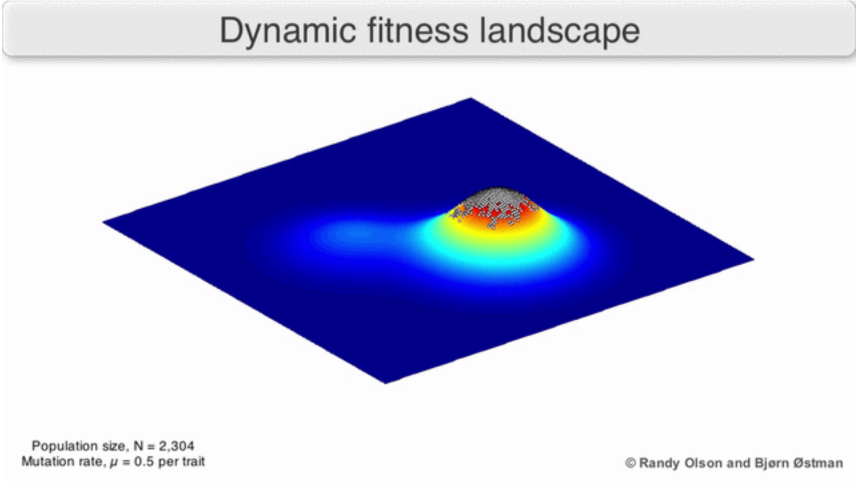
GA Numerical proof - Quod erat demonstratum



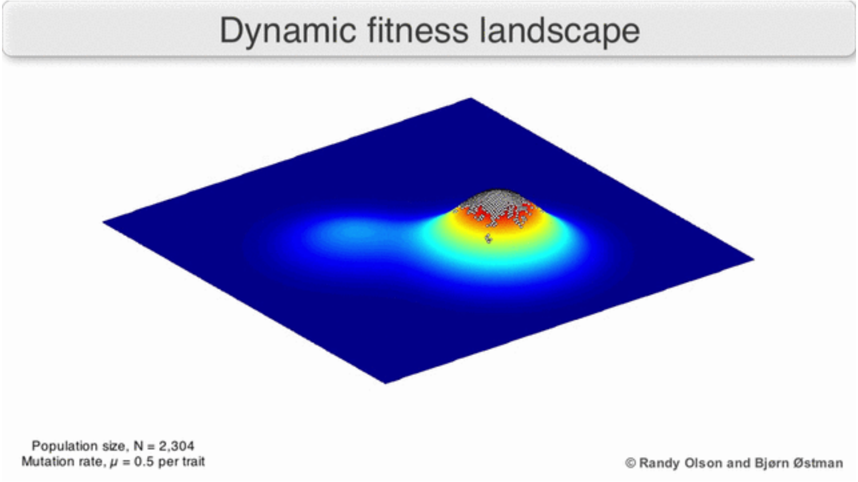
GA Numerical proof - Quod erat demonstratum



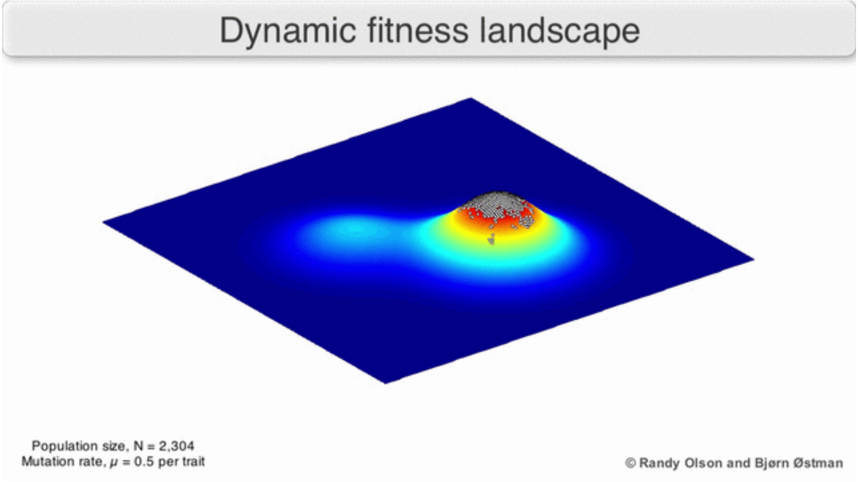
GA Numerical proof - Quod erat demonstratum



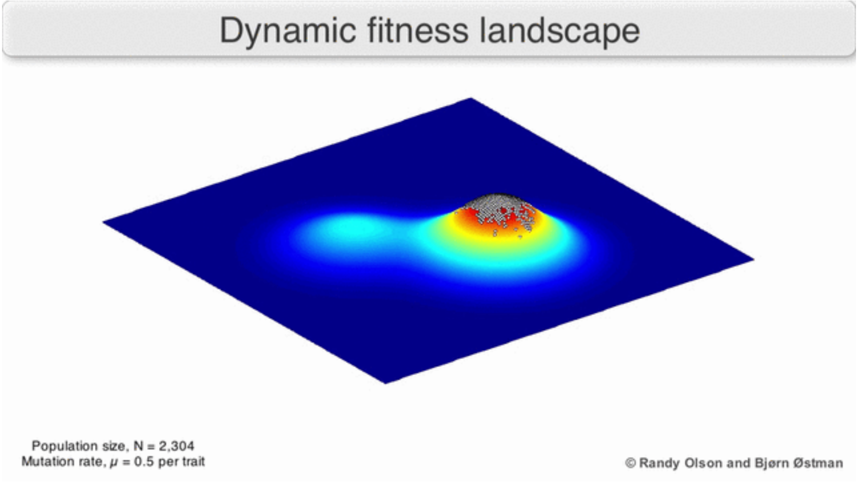
GA Numerical proof - Quod erat demonstratum



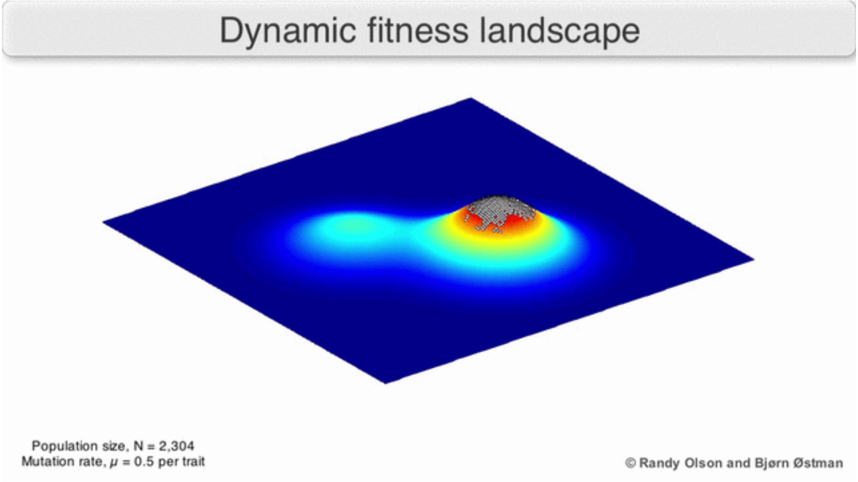
GA Numerical proof - Quod erat demonstratum



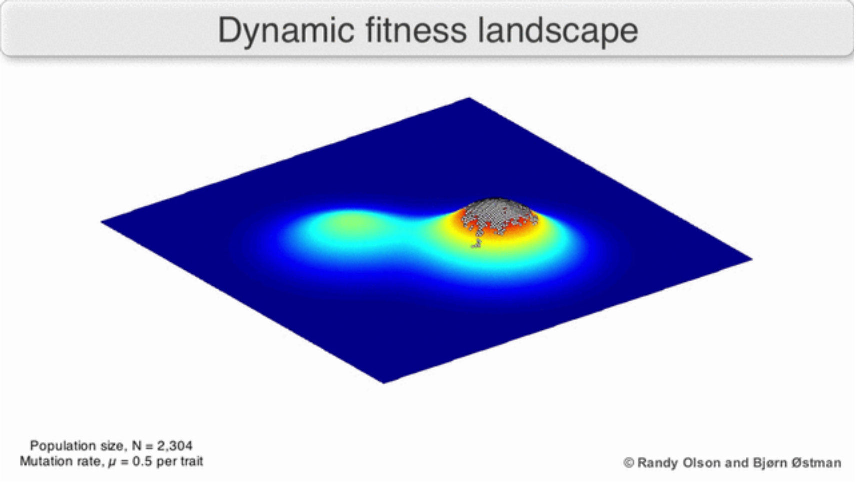
GA Numerical proof - Quod erat demonstratum



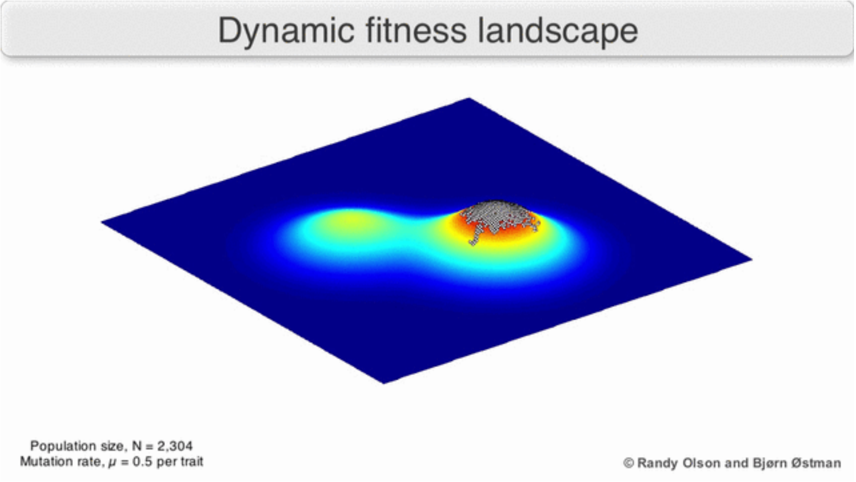
GA Numerical proof - Quod erat demonstratum



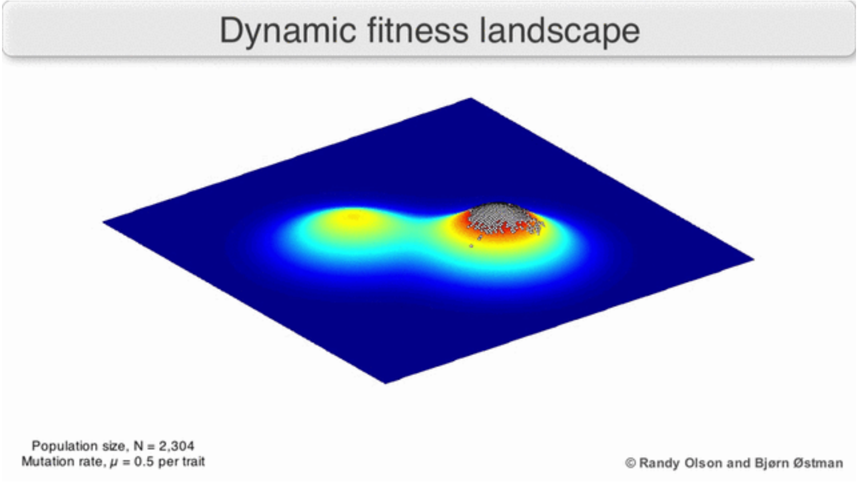
GA Numerical proof - Quod erat demonstratum



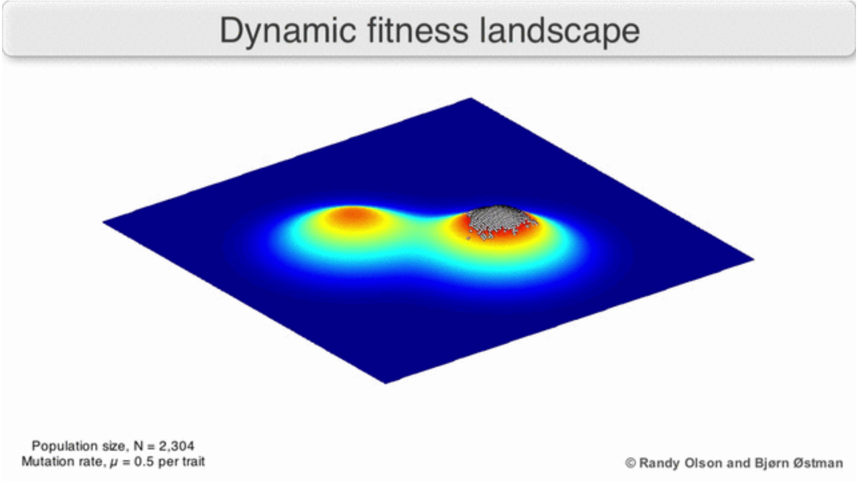
GA Numerical proof - Quod erat demonstratum



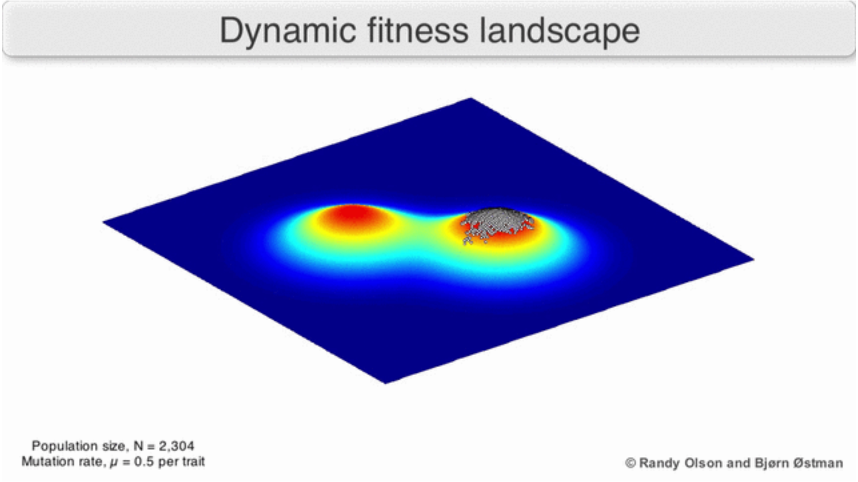
GA Numerical proof - Quod erat demonstratum



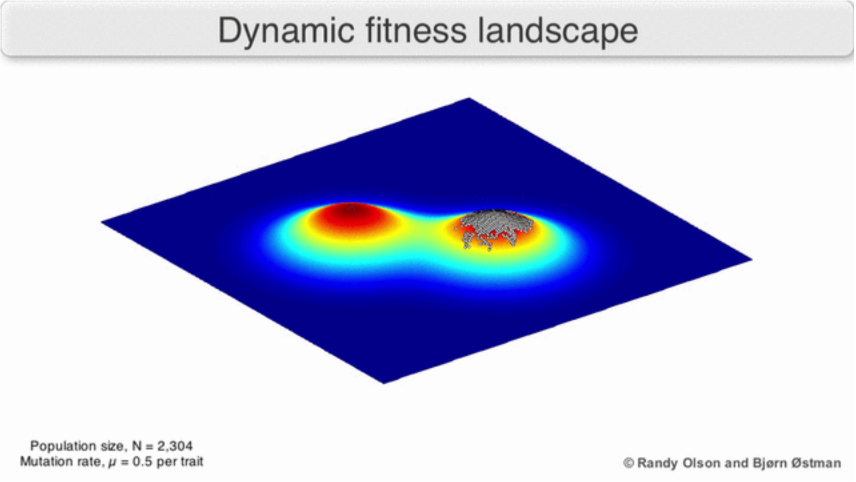
GA Numerical proof - Quod erat demonstratum



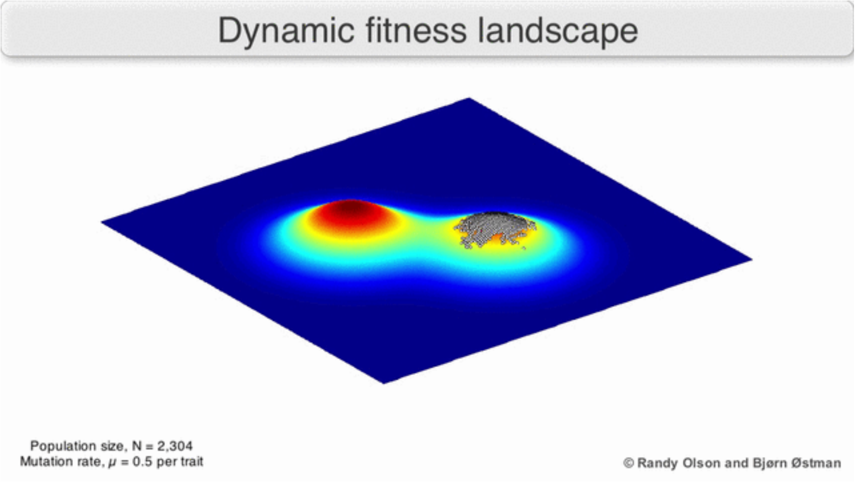
GA Numerical proof - Quod erat demonstratum



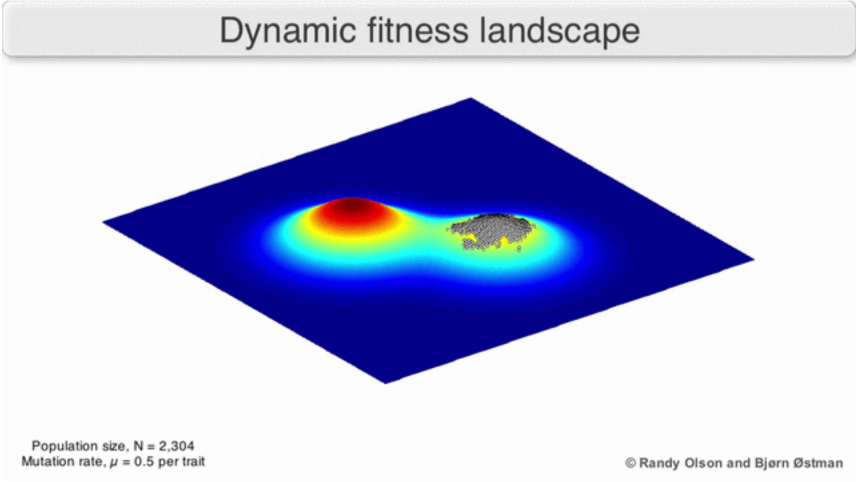
GA Numerical proof - Quod erat demonstratum



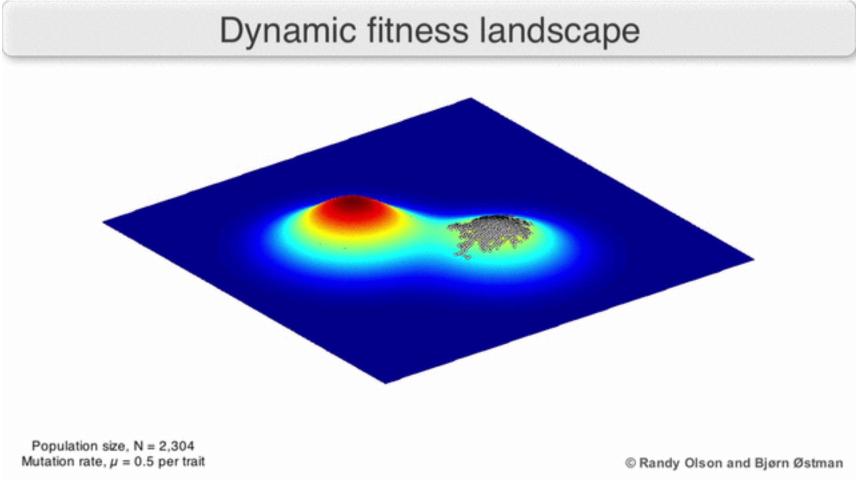
GA Numerical proof - Quod erat demonstratum



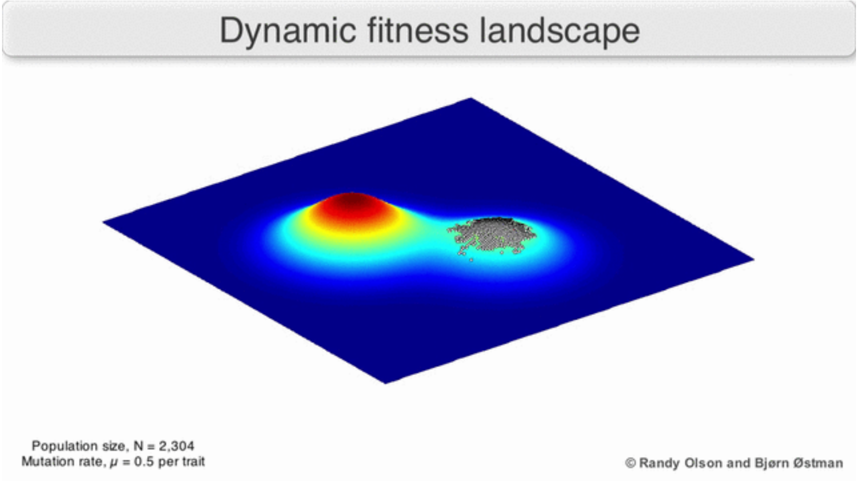
GA Numerical proof - Quod erat demonstratum



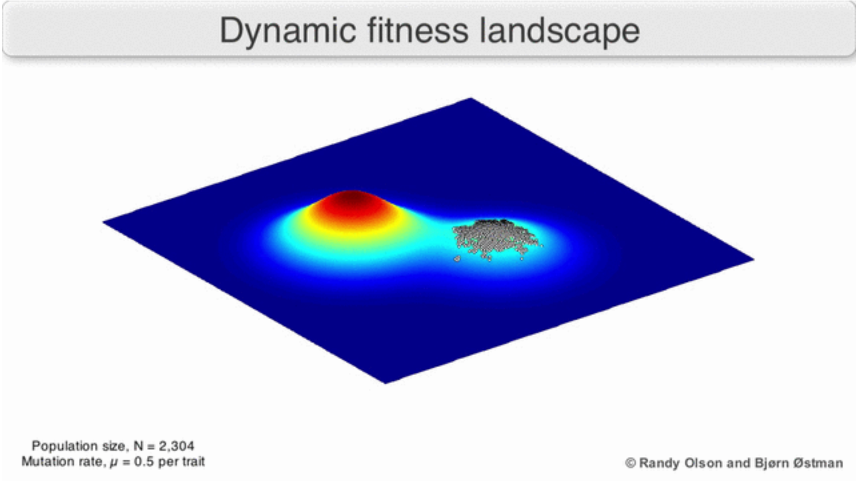
GA Numerical proof - Quod erat demonstratum



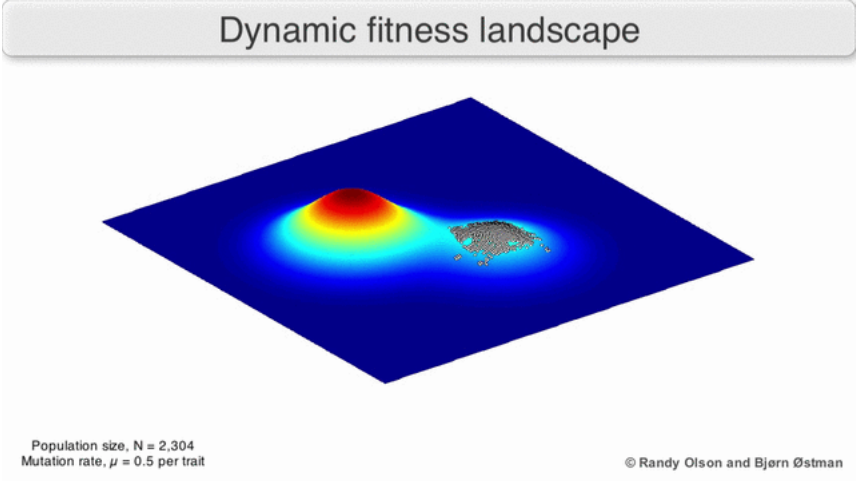
GA Numerical proof - Quod erat demonstratum



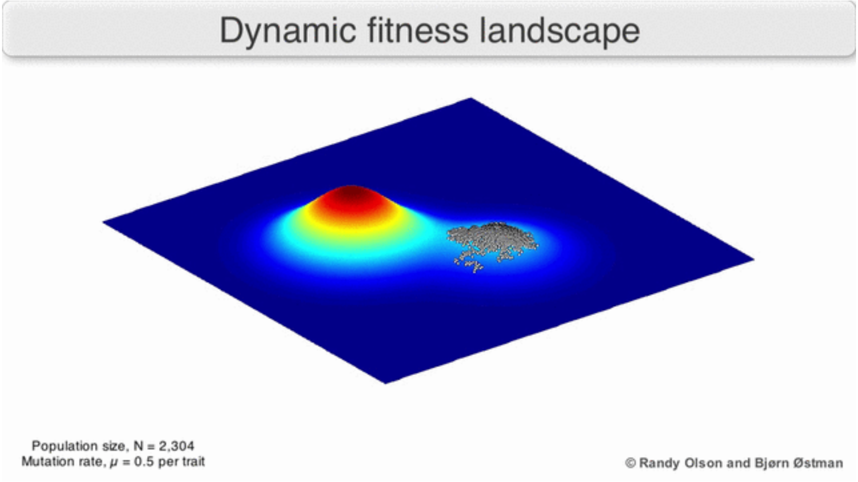
GA Numerical proof - Quod erat demonstratum



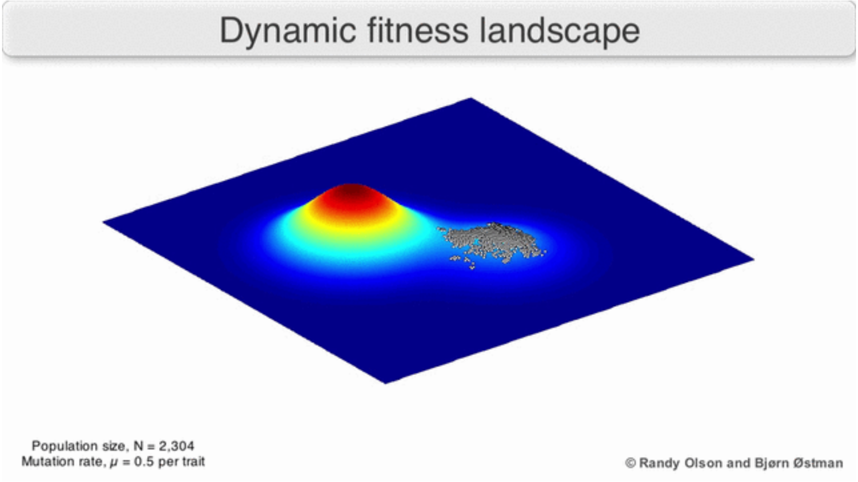
GA Numerical proof - Quod erat demonstratum



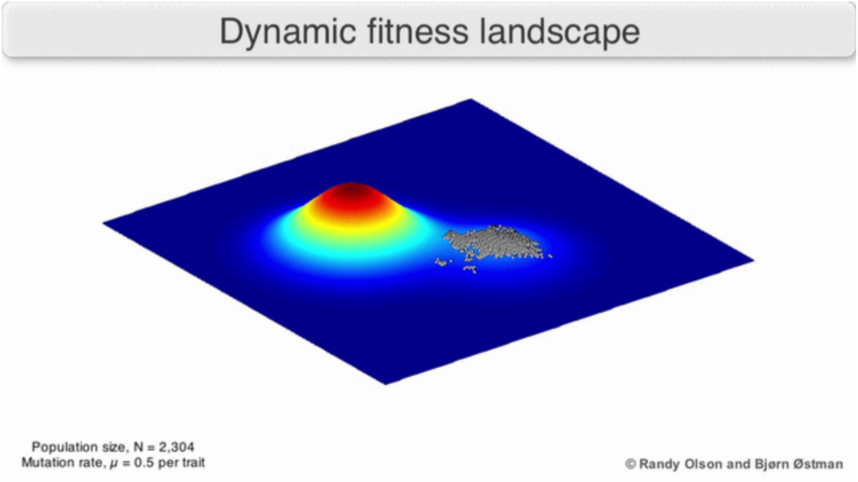
GA Numerical proof - Quod erat demonstratum



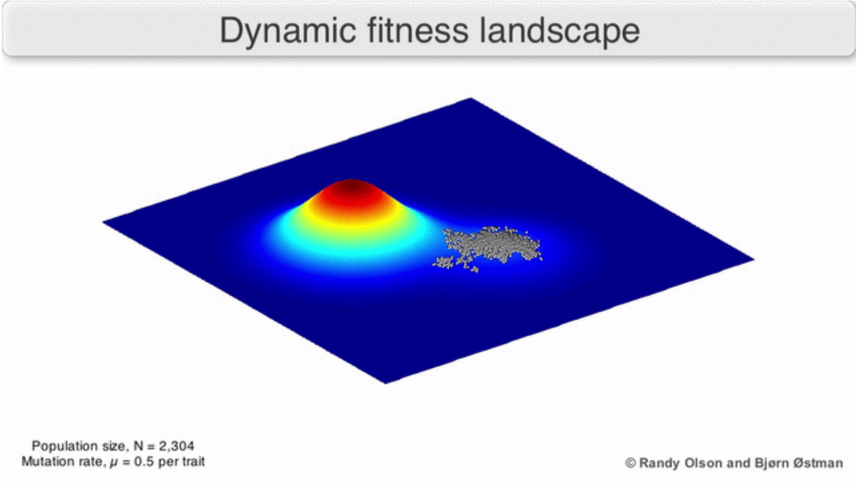
GA Numerical proof - Quod erat demonstratum



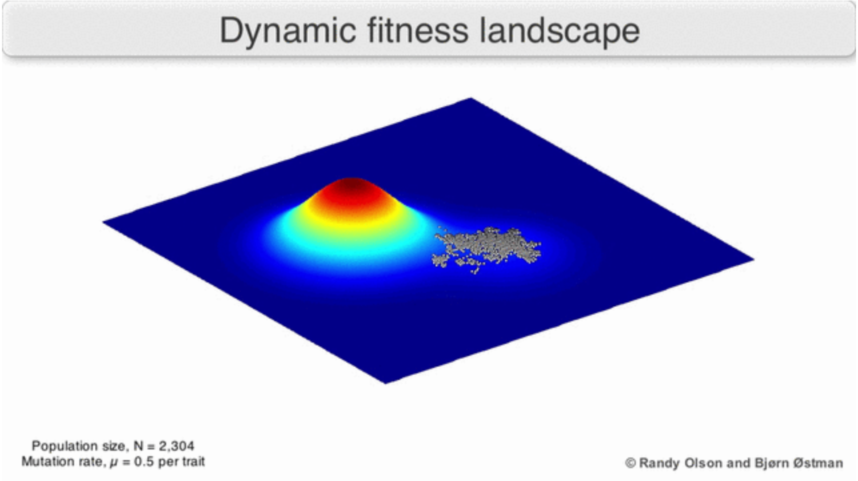
GA Numerical proof - Quod erat demonstratum



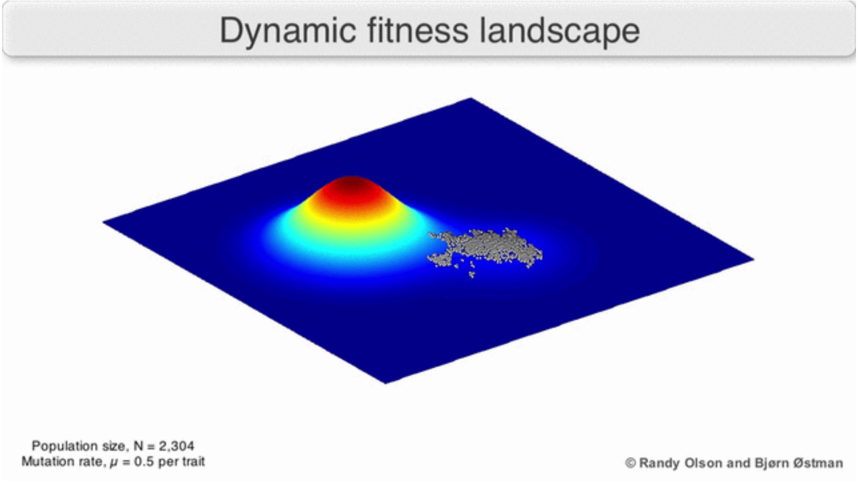
GA Numerical proof - Quod erat demonstratum



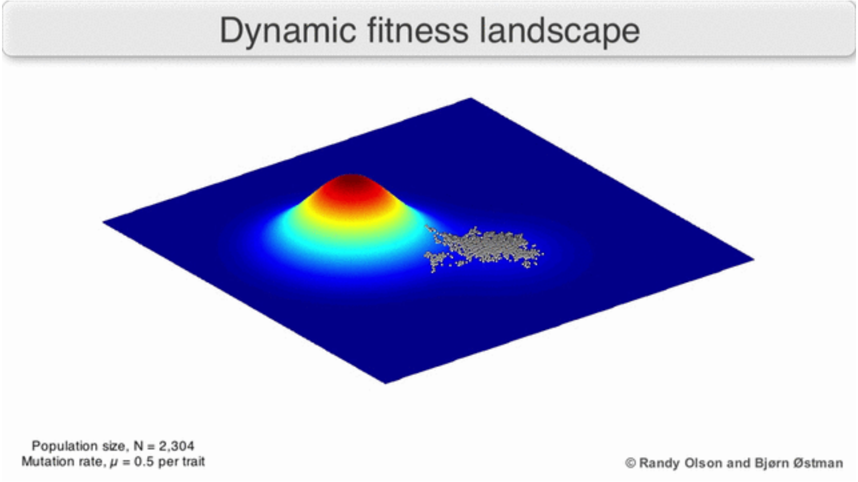
GA Numerical proof - Quod erat demonstratum



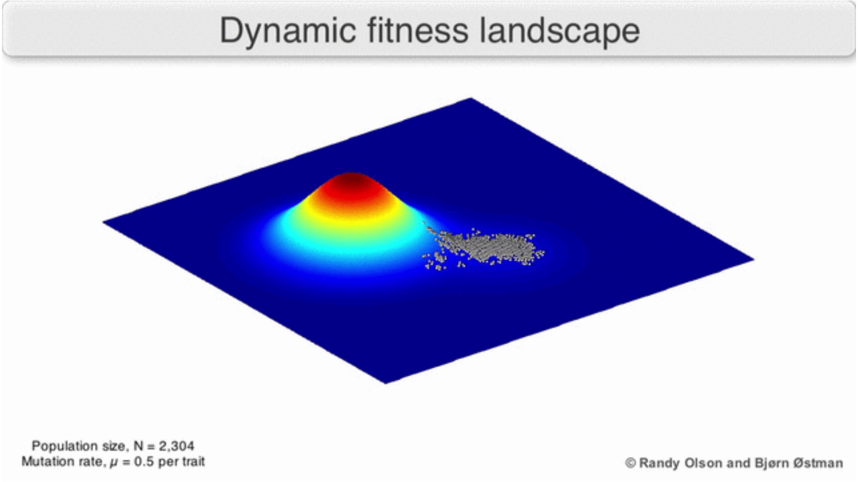
GA Numerical proof - Quod erat demonstratum



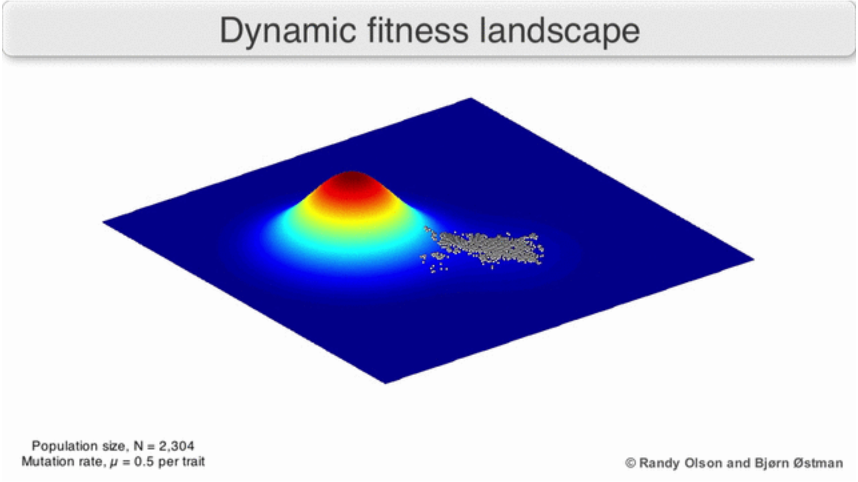
GA Numerical proof - Quod erat demonstratum



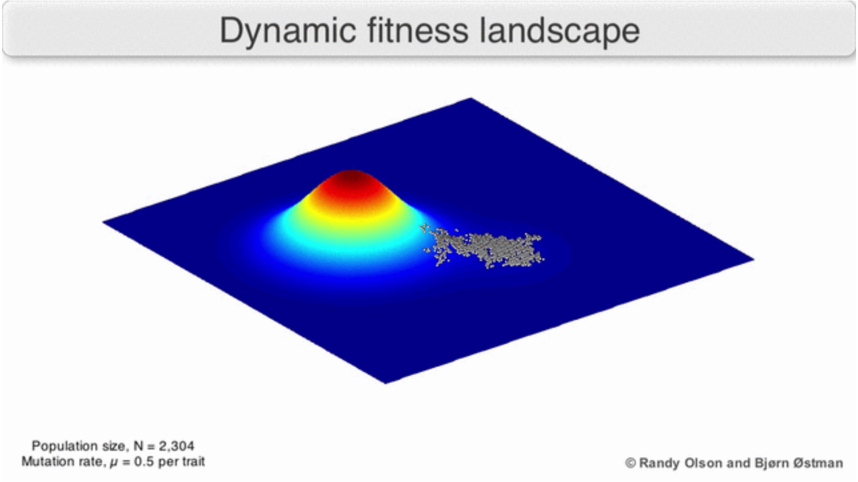
GA Numerical proof - Quod erat demonstratum



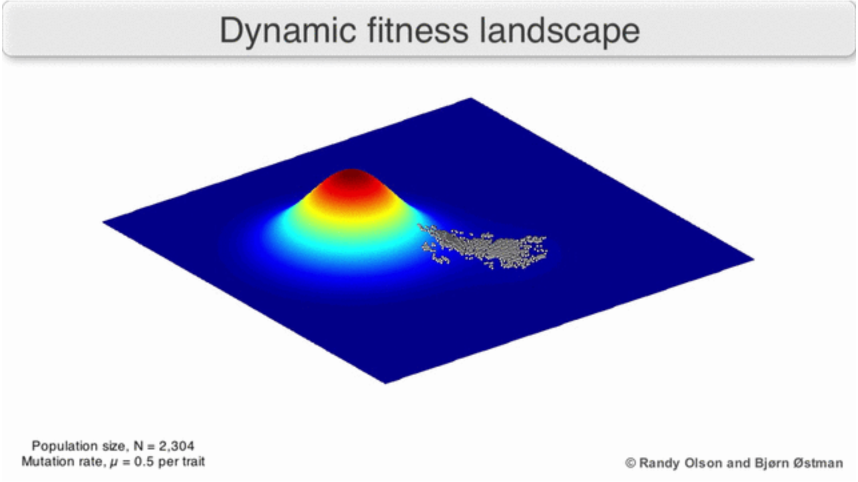
GA Numerical proof - Quod erat demonstratum



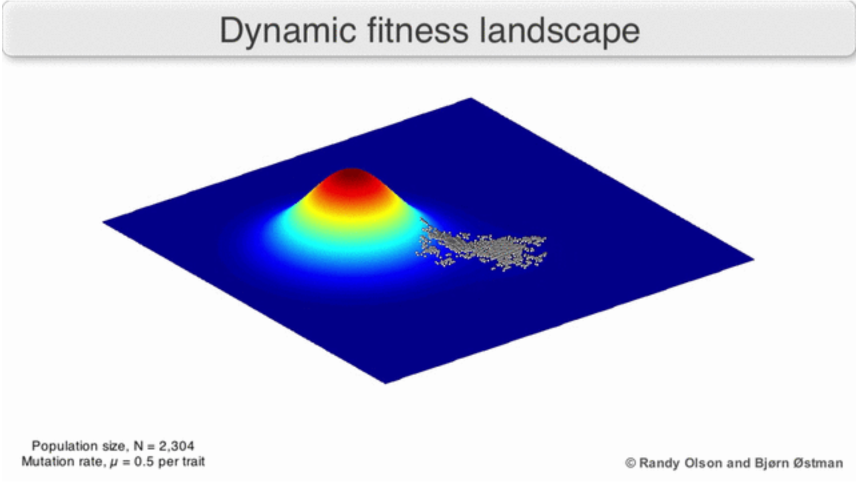
GA Numerical proof - Quod erat demonstratum



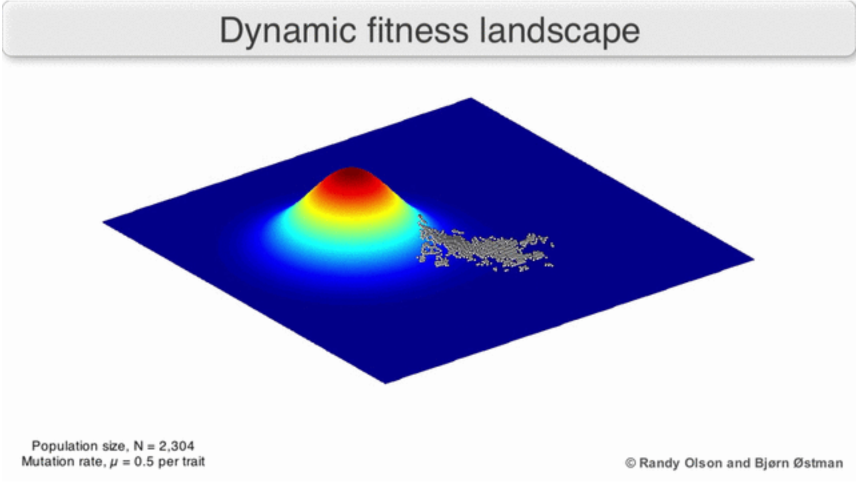
GA Numerical proof - Quod erat demonstratum



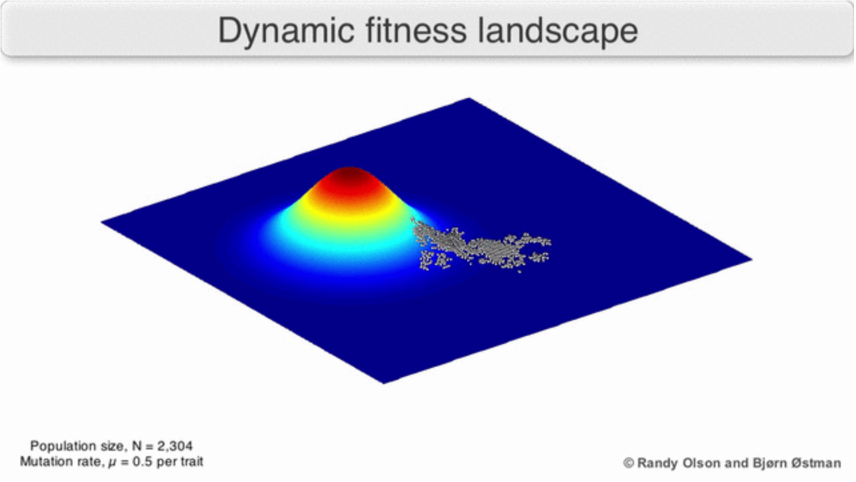
GA Numerical proof - Quod erat demonstratum



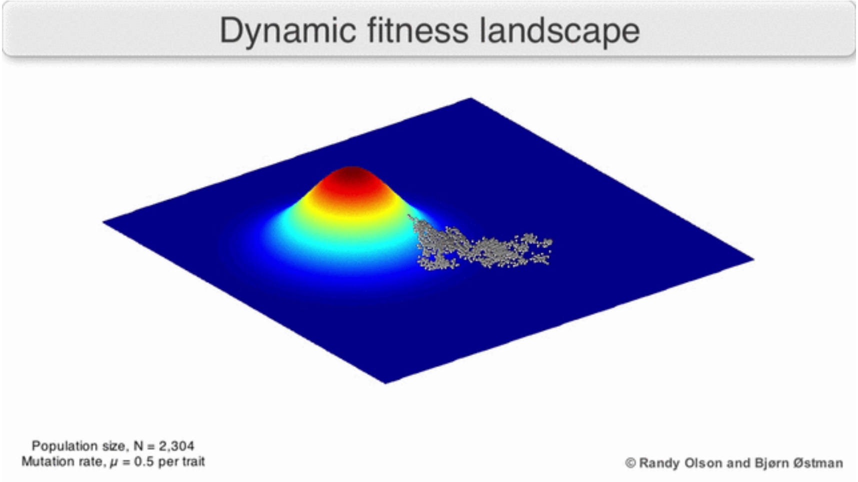
GA Numerical proof - Quod erat demonstratum



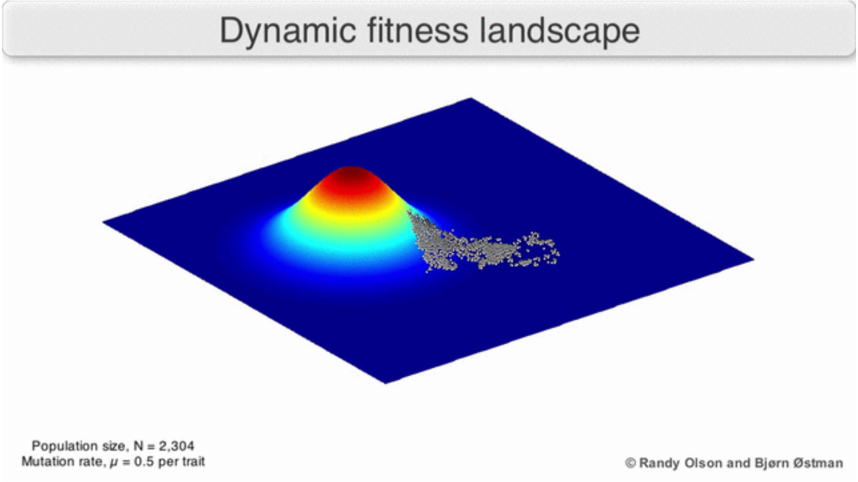
GA Numerical proof - Quod erat demonstratum



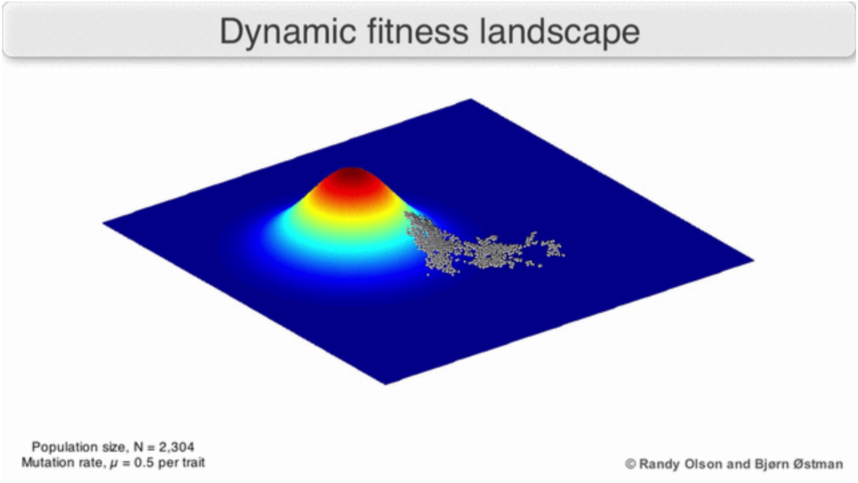
GA Numerical proof - Quod erat demonstratum



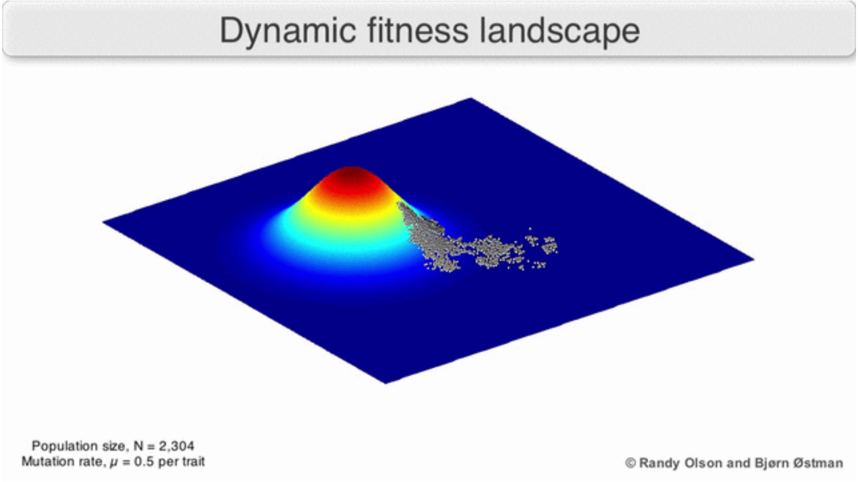
GA Numerical proof - Quod erat demonstratum



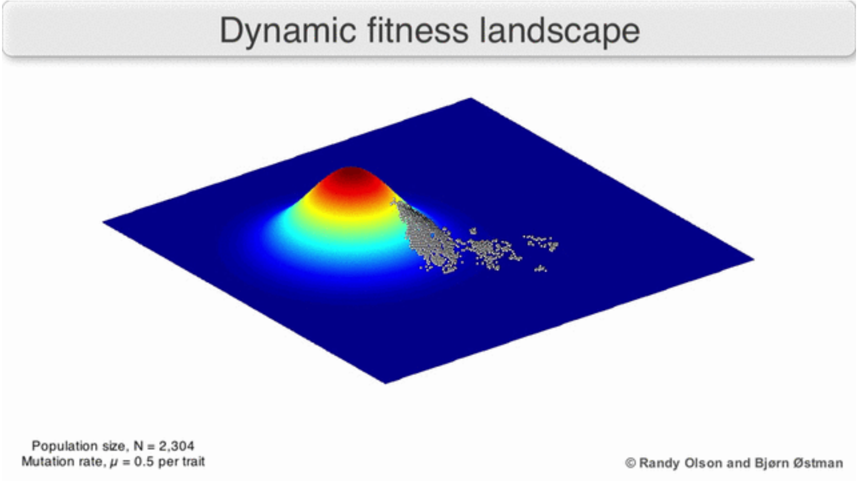
GA Numerical proof - Quod erat demonstratum



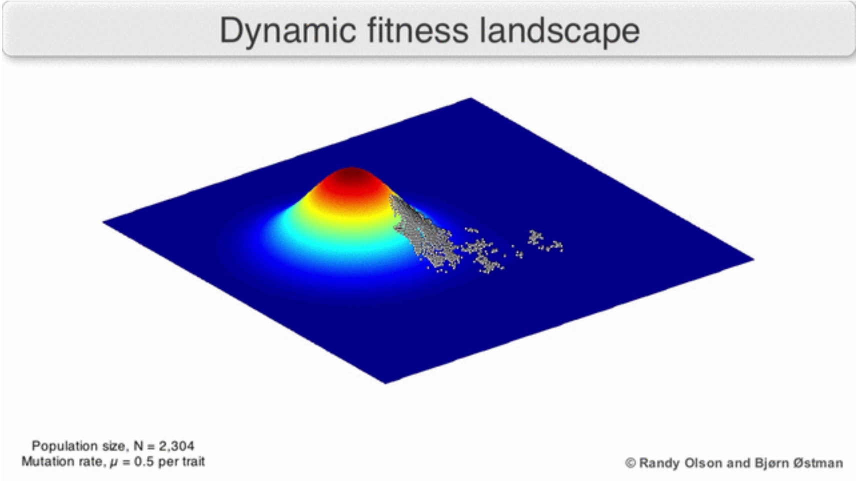
GA Numerical proof - Quod erat demonstratum



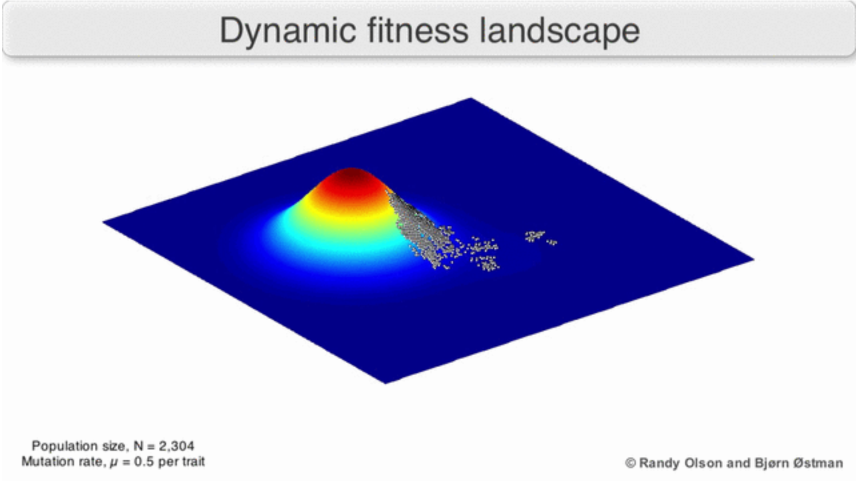
GA Numerical proof - Quod erat demonstratum



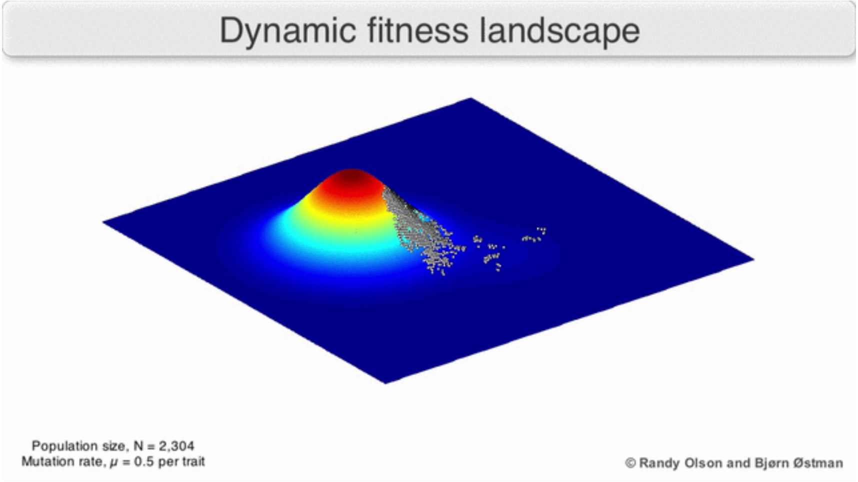
GA Numerical proof - Quod erat demonstratum



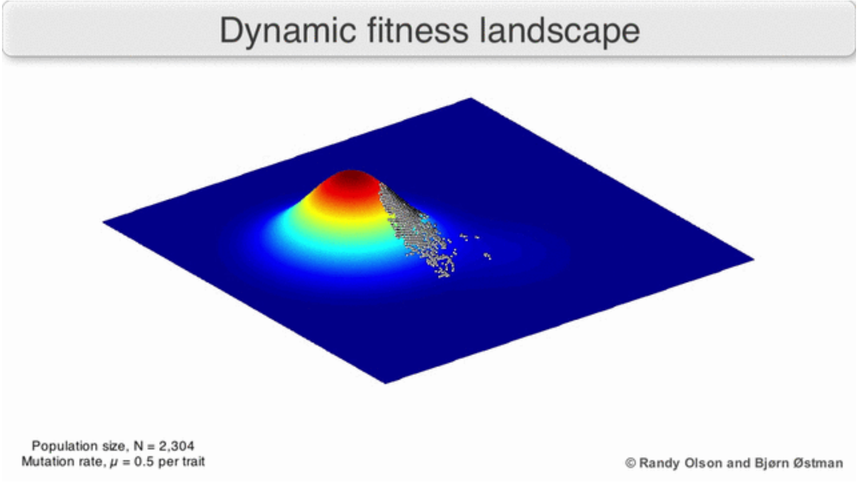
GA Numerical proof - Quod erat demonstratum



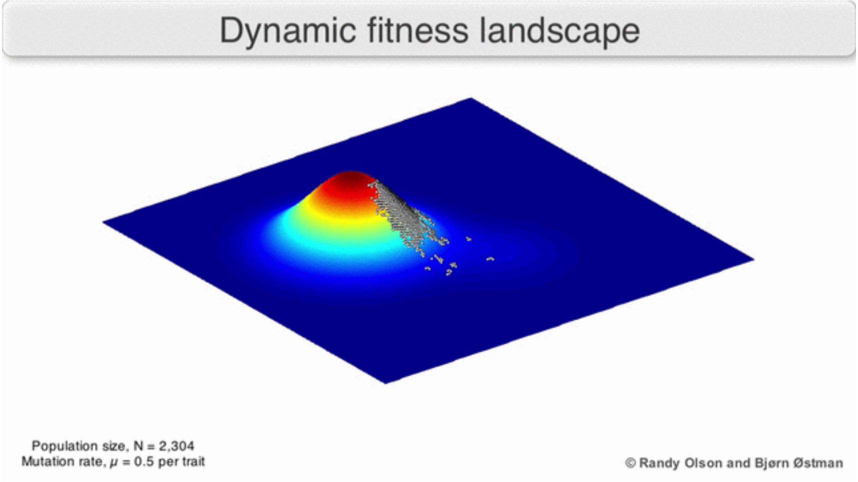
GA Numerical proof - Quod erat demonstratum



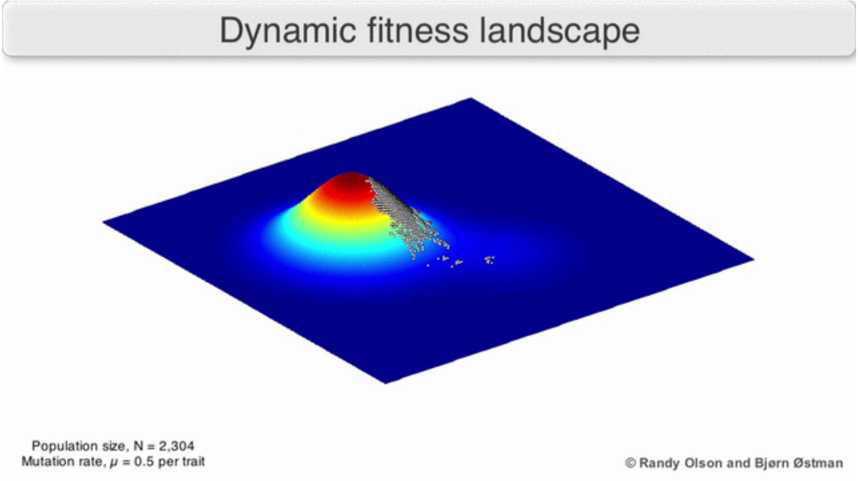
GA Numerical proof - Quod erat demonstratum



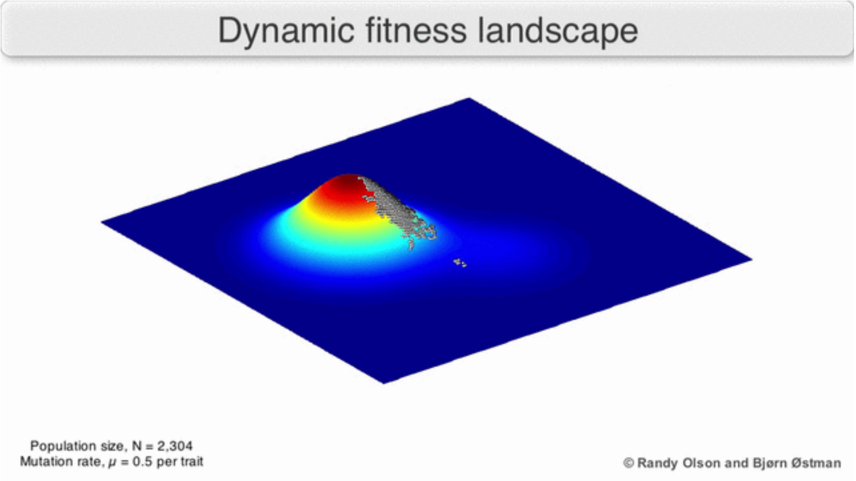
GA Numerical proof - Quod erat demonstratum



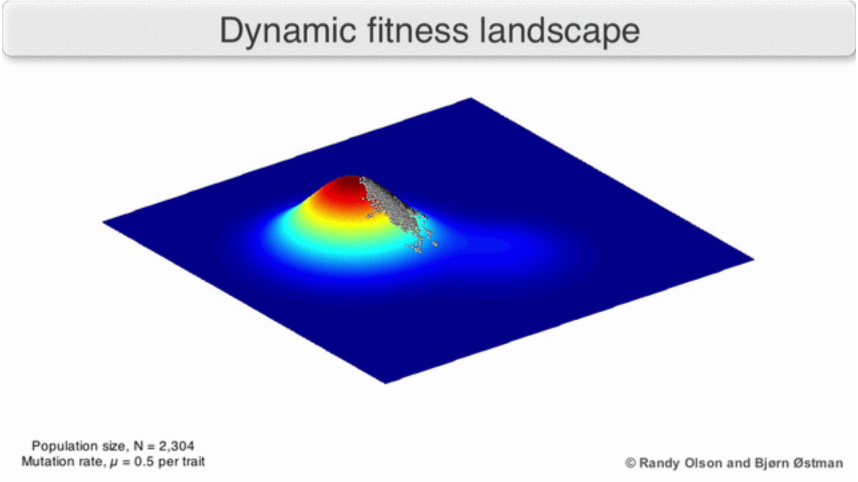
GA Numerical proof - Quod erat demonstratum



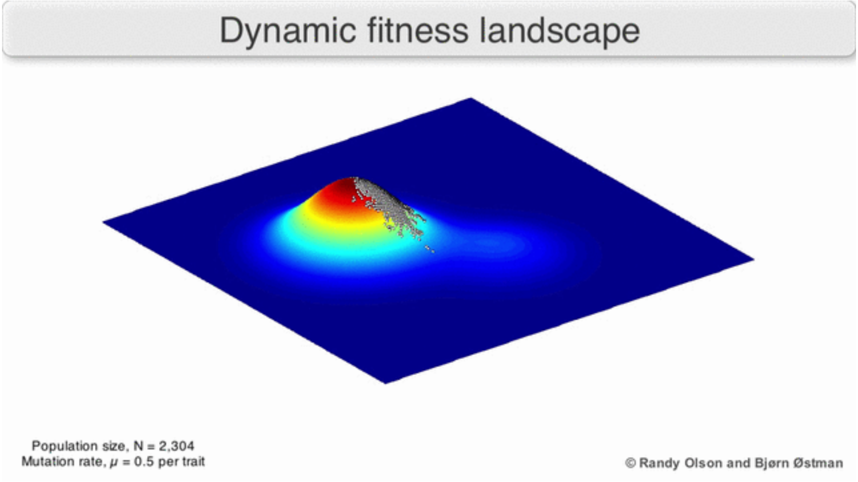
GA Numerical proof - Quod erat demonstratum



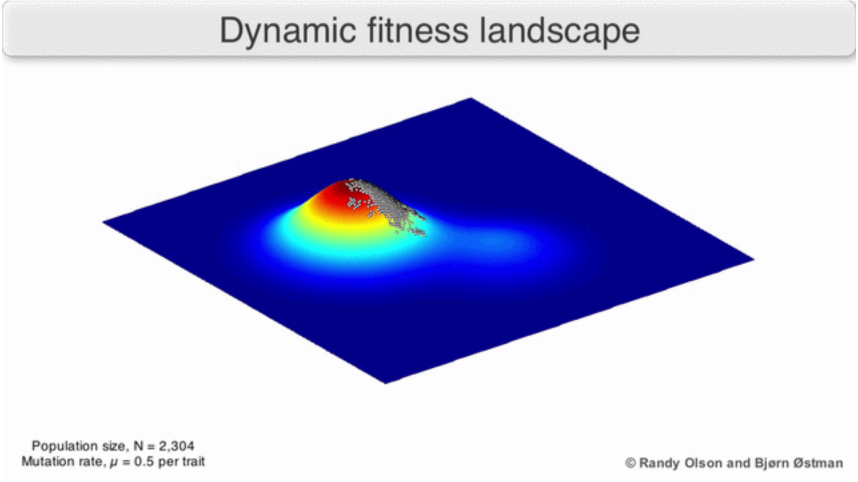
GA Numerical proof - Quod erat demonstratum



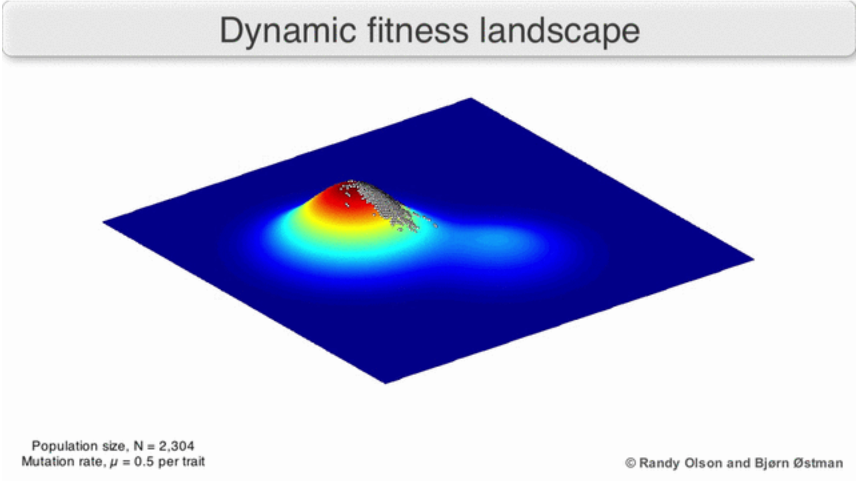
GA Numerical proof - Quod erat demonstratum



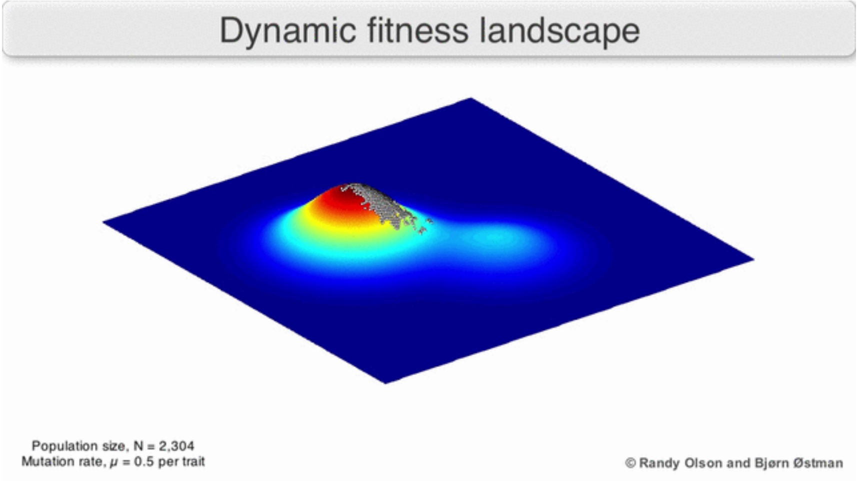
GA Numerical proof - Quod erat demonstratum



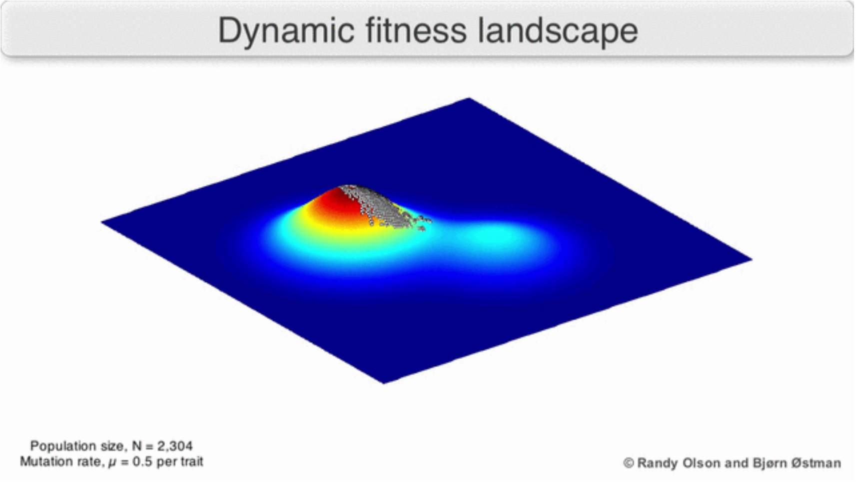
GA Numerical proof - Quod erat demonstratum



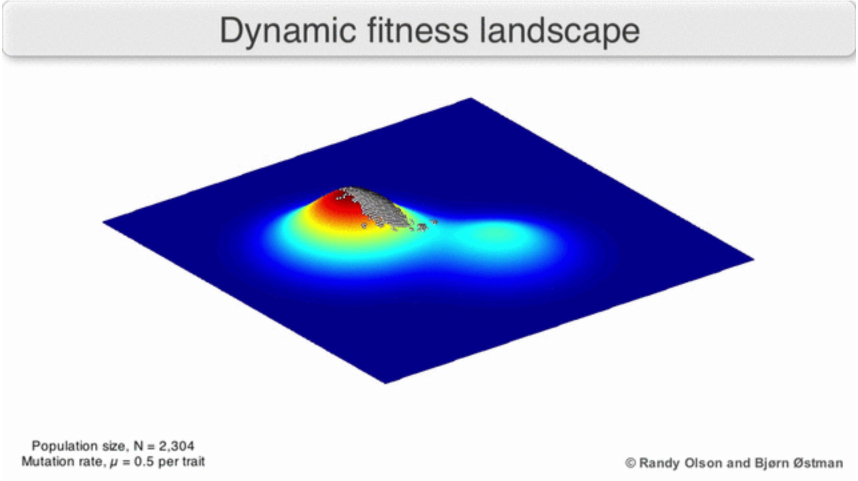
GA Numerical proof - Quod erat demonstratum



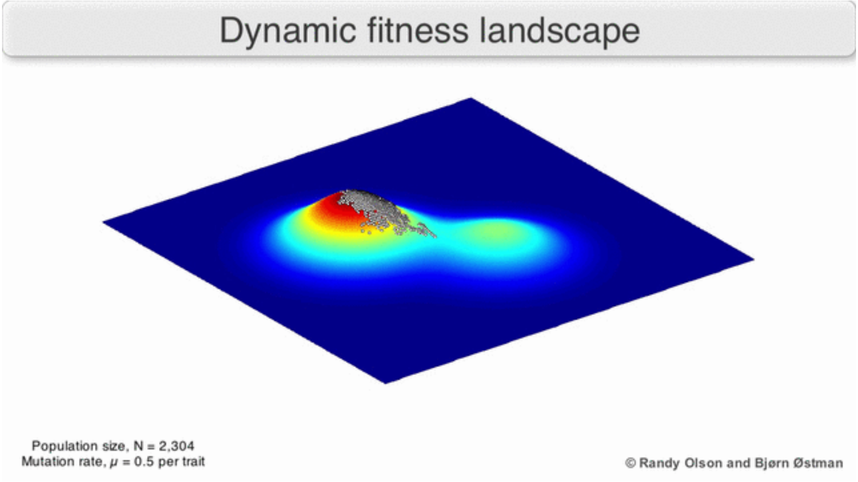
GA Numerical proof - Quod erat demonstratum



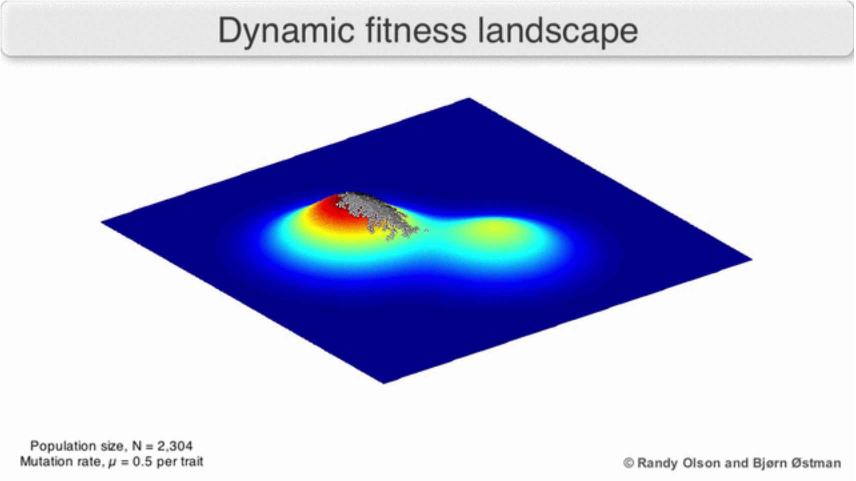
GA Numerical proof - Quod erat demonstratum



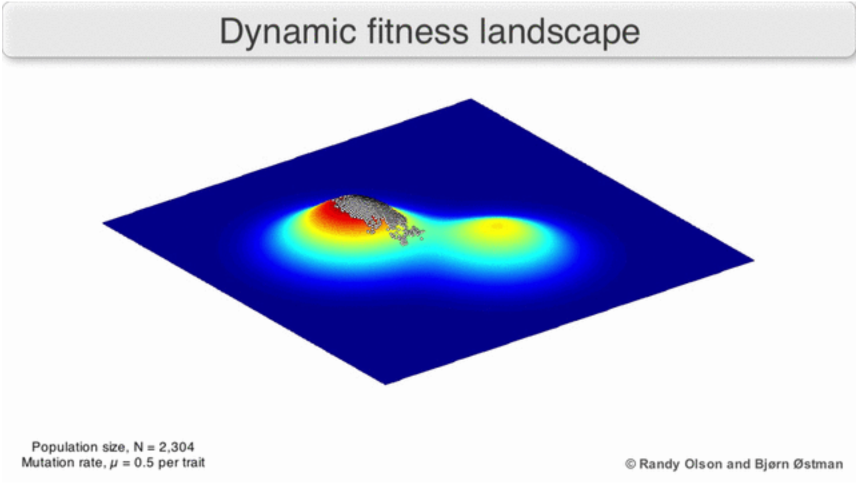
GA Numerical proof - Quod erat demonstratum



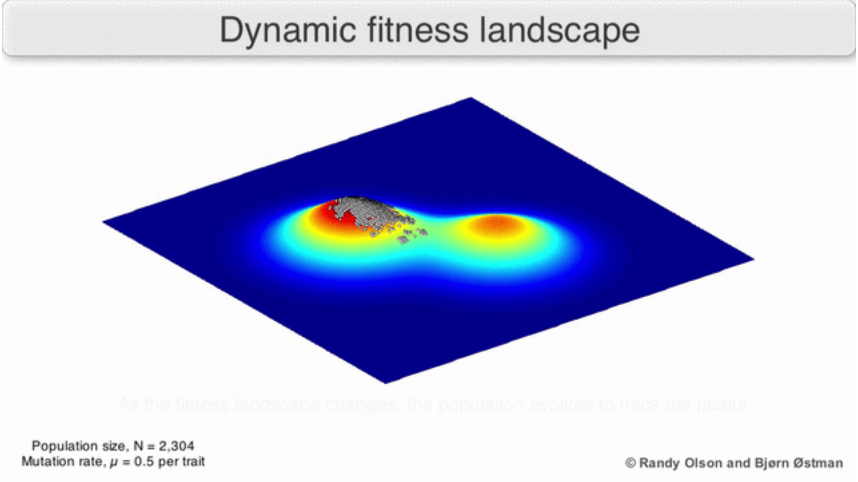
GA Numerical proof - Quod erat demonstratum



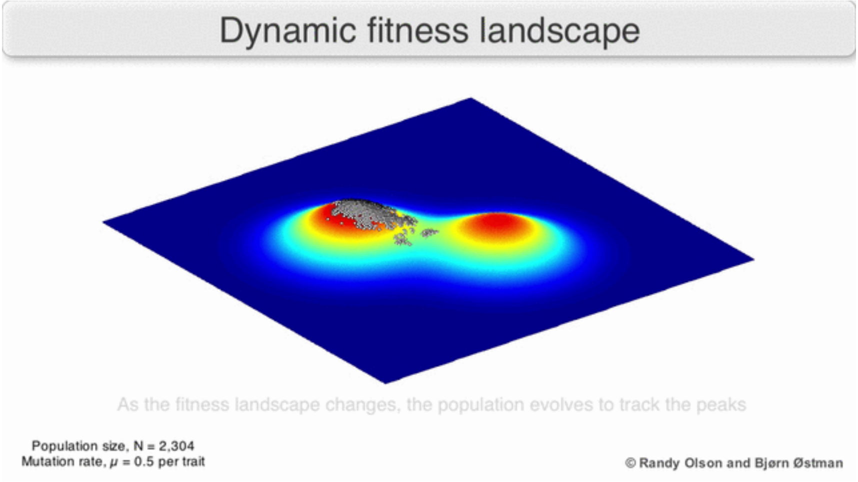
GA Numerical proof - Quod erat demonstratum



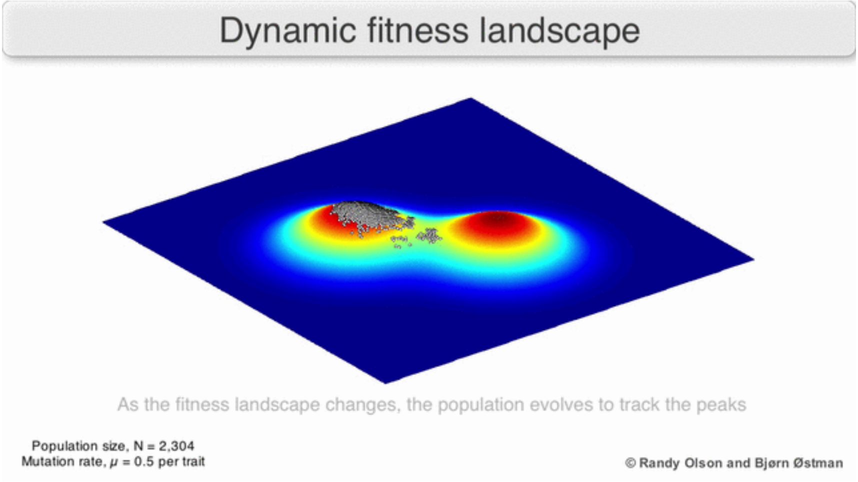
GA Numerical proof - Quod erat demonstratum



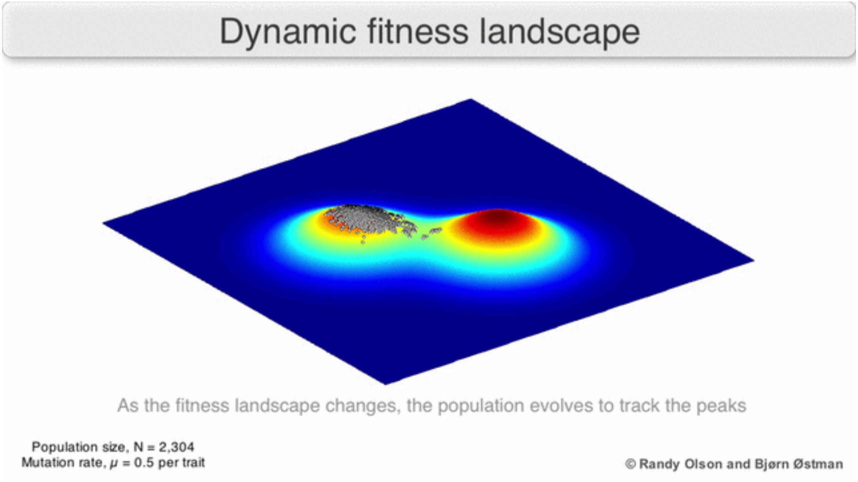
GA Numerical proof - Quod erat demonstratum



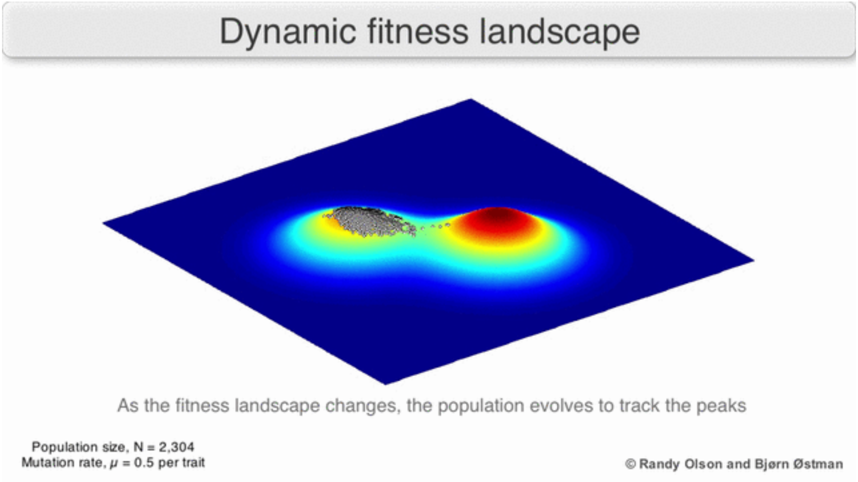
GA Numerical proof - Quod erat demonstratum



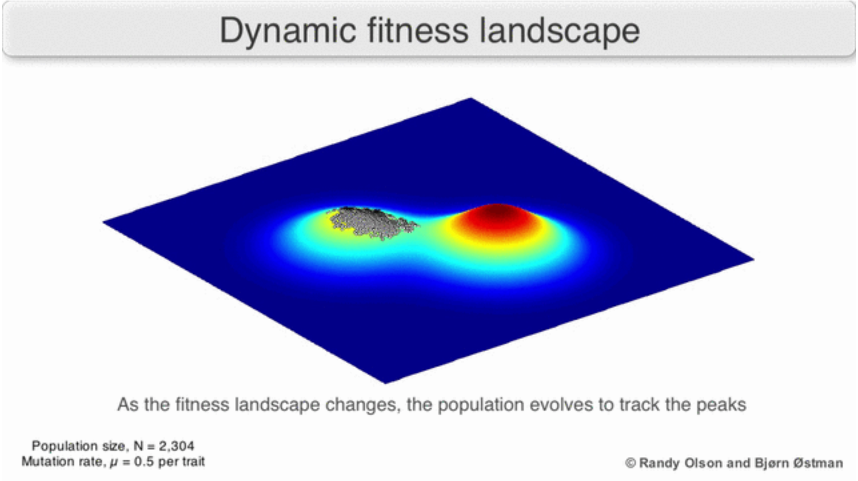
GA Numerical proof - Quod erat demonstratum



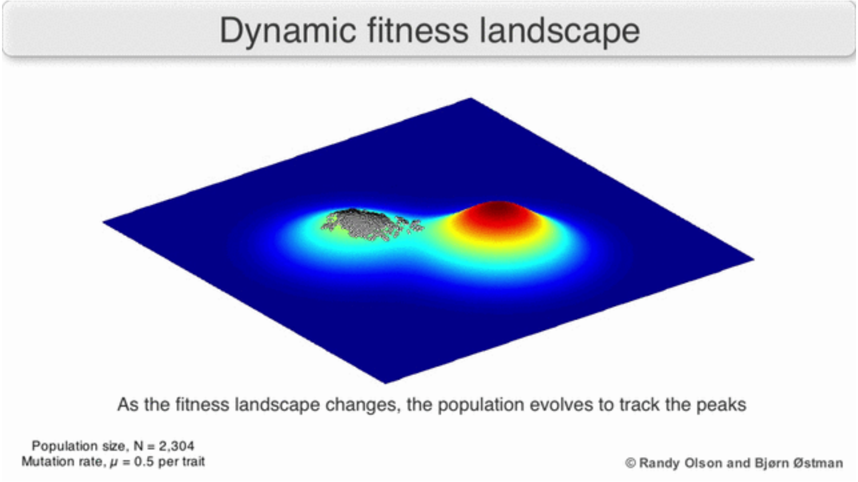
GA Numerical proof - Quod erat demonstratum



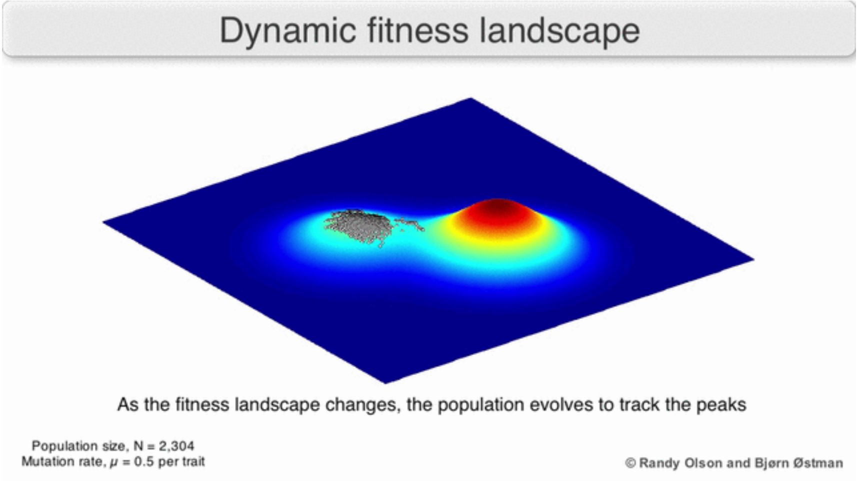
GA Numerical proof - Quod erat demonstratum



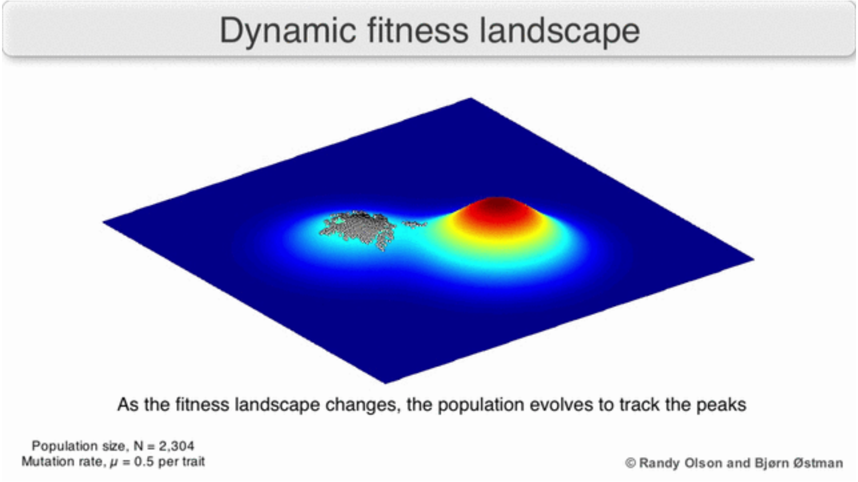
GA Numerical proof - Quod erat demonstratum



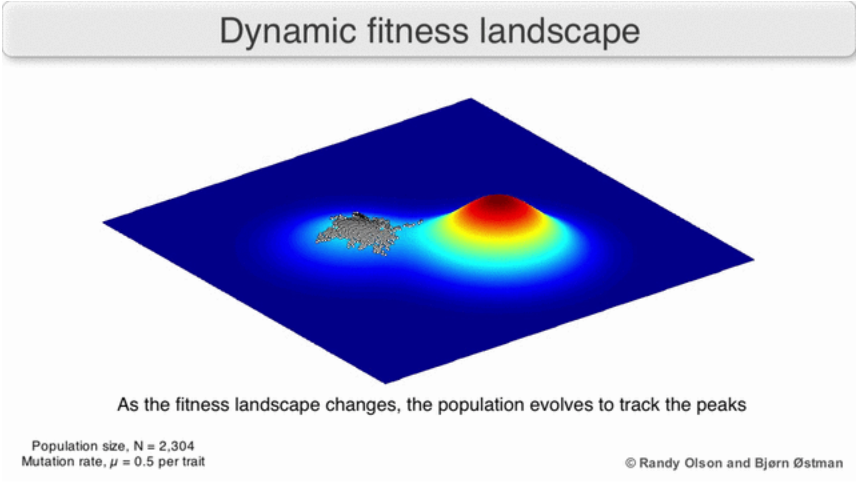
GA Numerical proof - Quod erat demonstratum



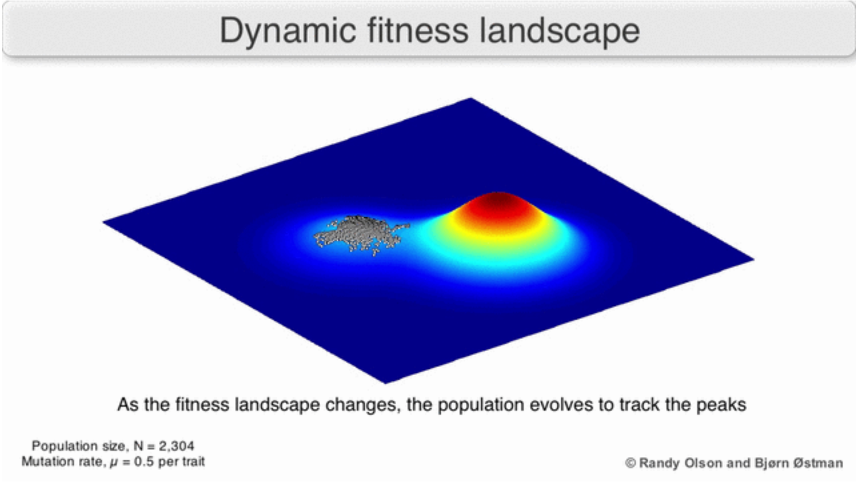
GA Numerical proof - Quod erat demonstratum



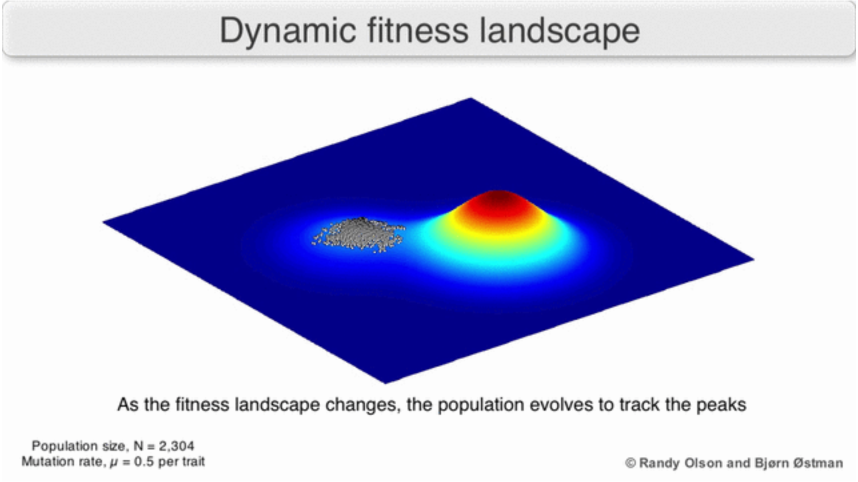
GA Numerical proof - Quod erat demonstratum



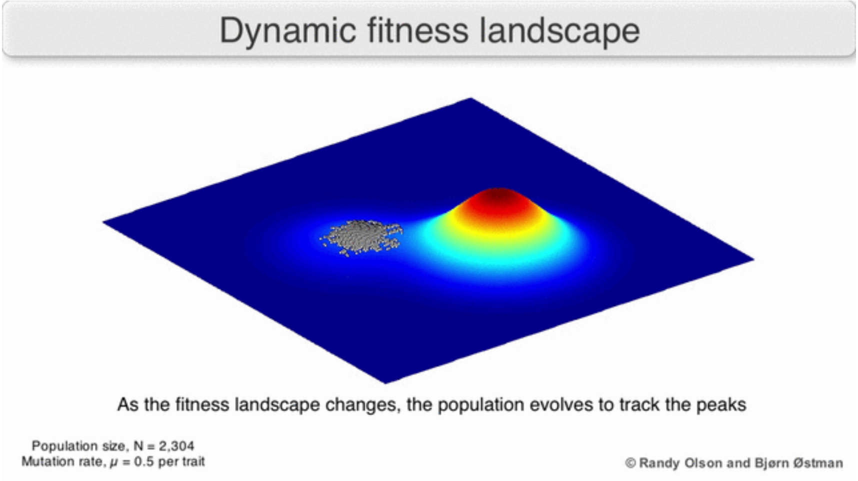
GA Numerical proof - Quod erat demonstratum



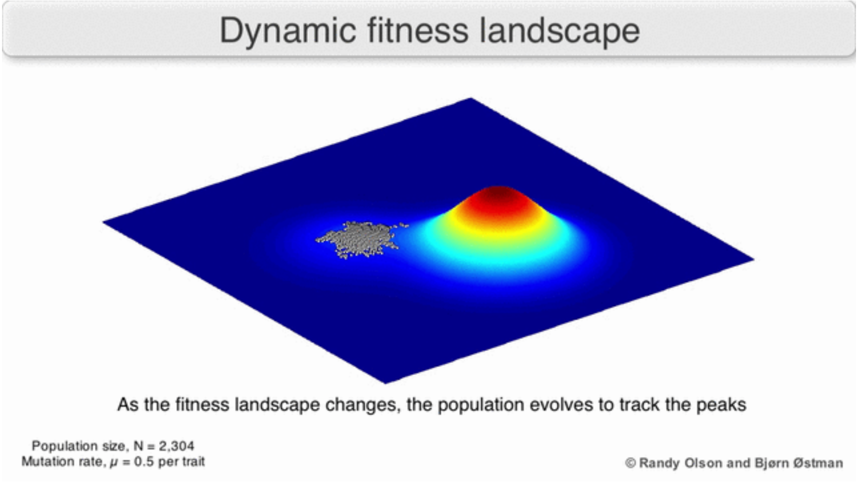
GA Numerical proof - Quod erat demonstratum



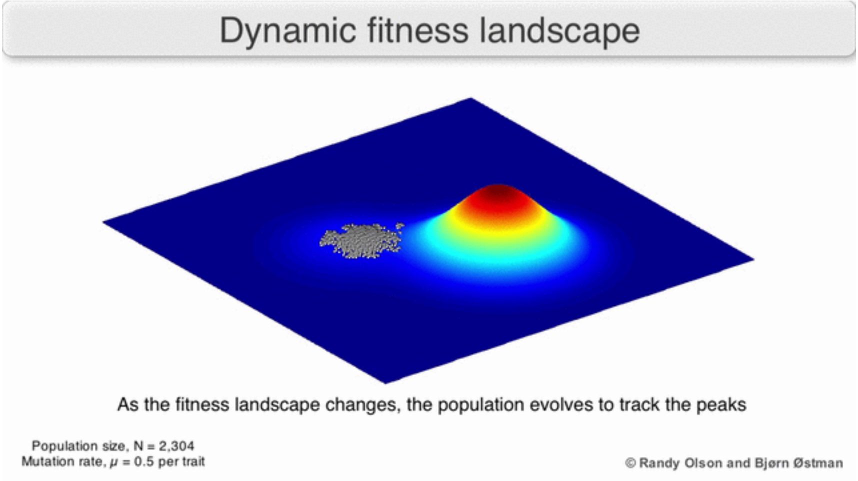
GA Numerical proof - Quod erat demonstratum



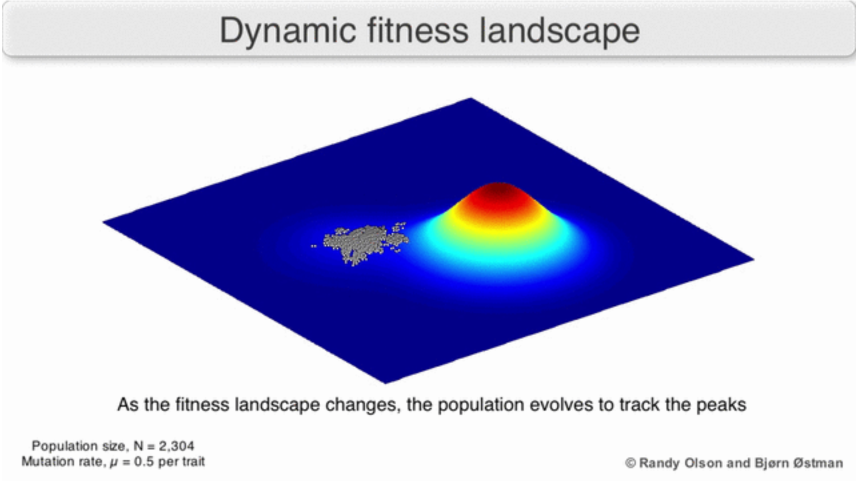
GA Numerical proof - Quod erat demonstratum



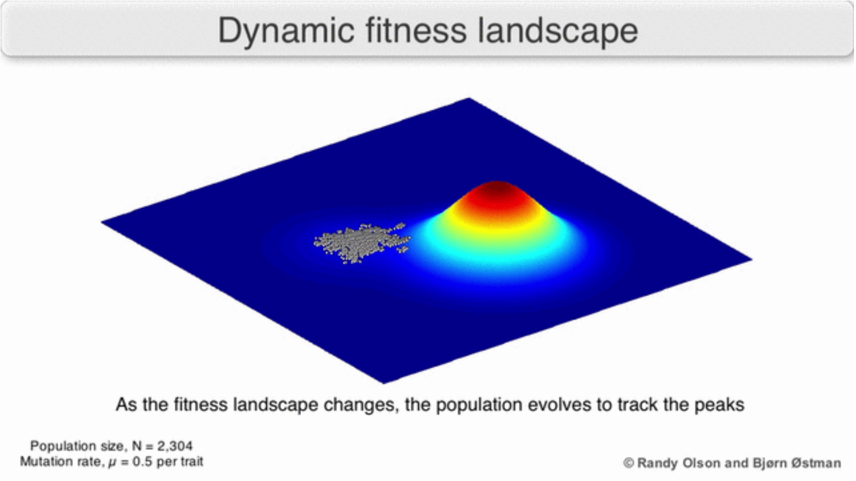
GA Numerical proof - Quod erat demonstratum



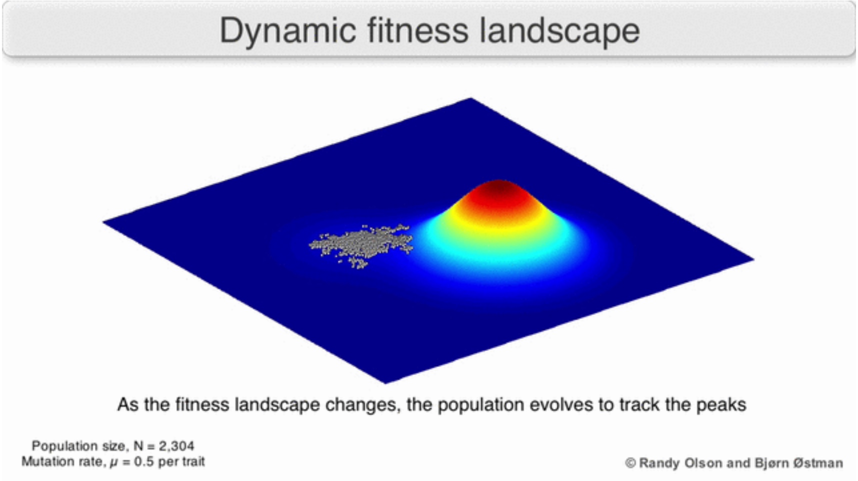
GA Numerical proof - Quod erat demonstratum



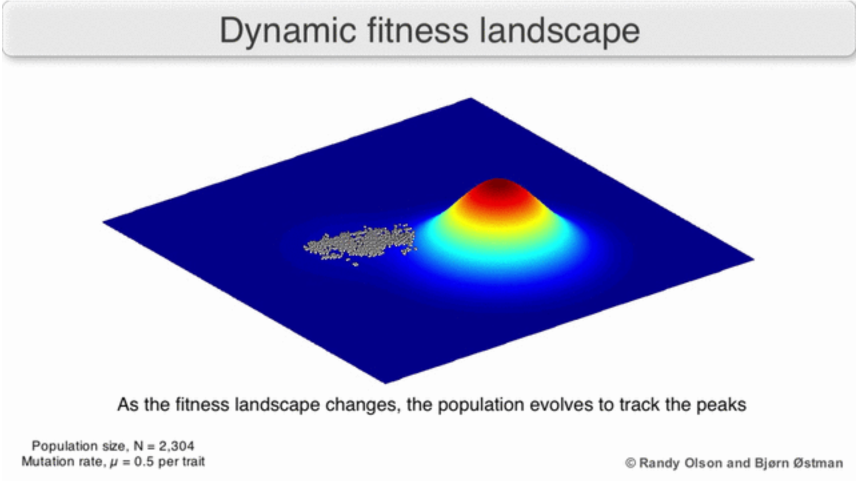
GA Numerical proof - Quod erat demonstratum



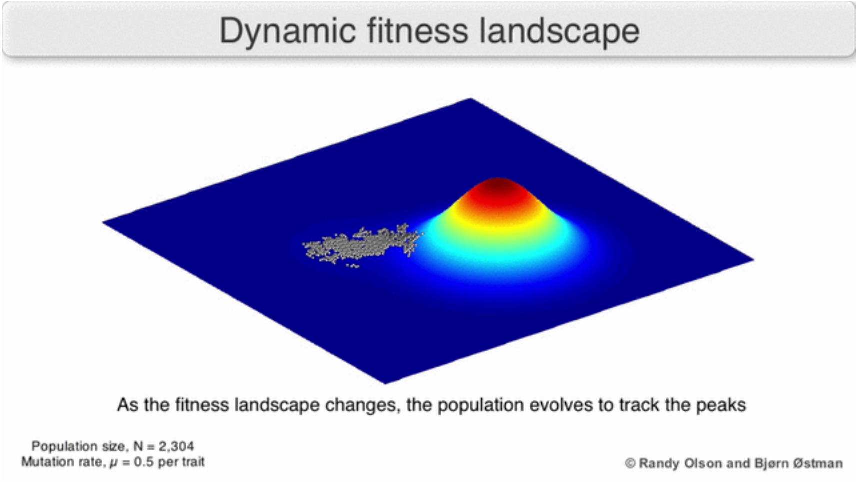
GA Numerical proof - Quod erat demonstratum



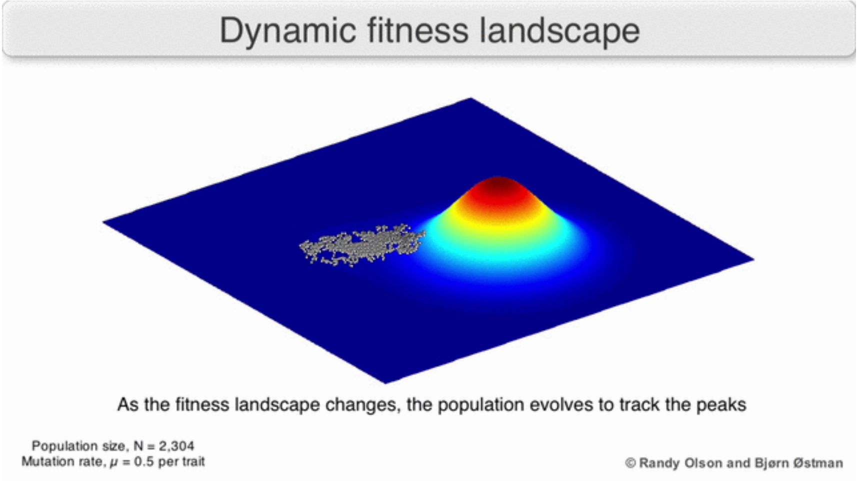
GA Numerical proof - Quod erat demonstratum



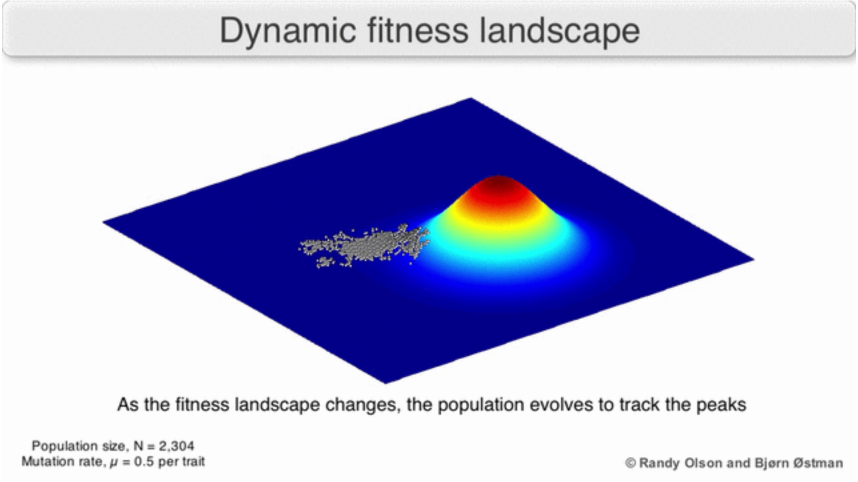
GA Numerical proof - Quod erat demonstratum



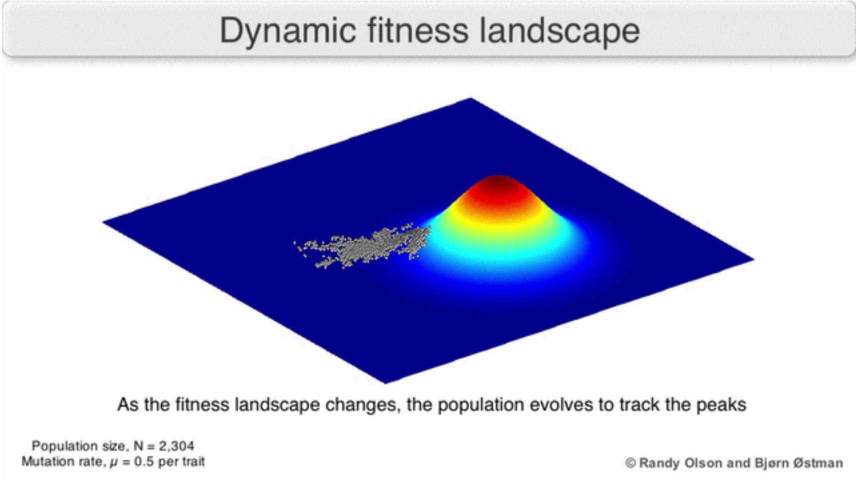
GA Numerical proof - Quod erat demonstratum



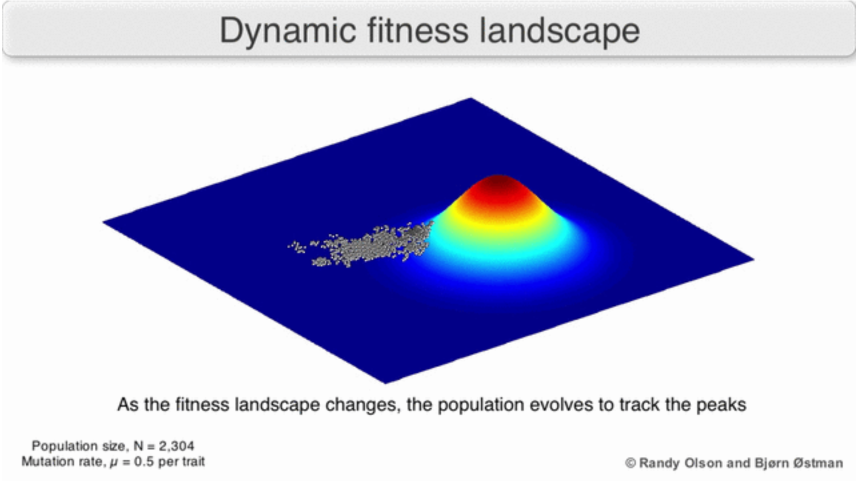
GA Numerical proof - Quod erat demonstratum



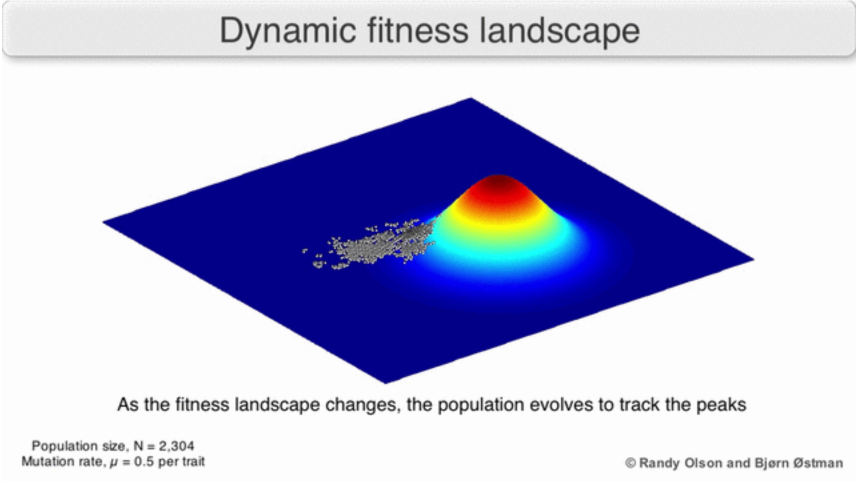
GA Numerical proof - Quod erat demonstratum



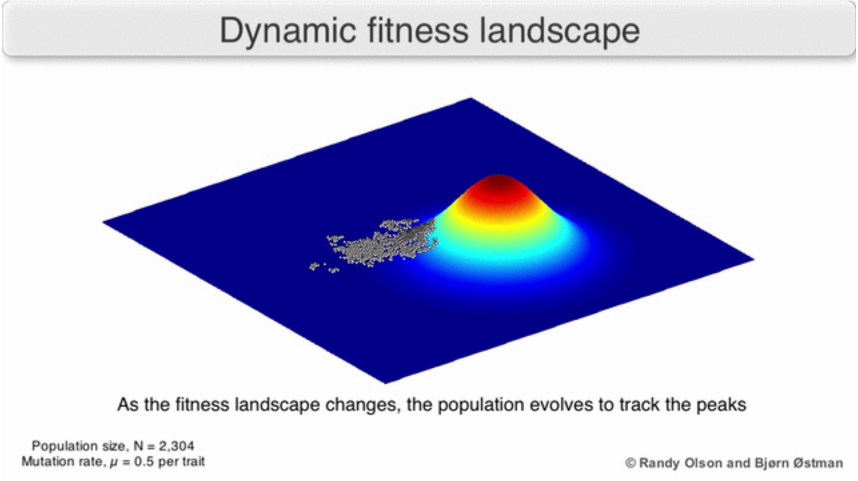
GA Numerical proof - Quod erat demonstratum



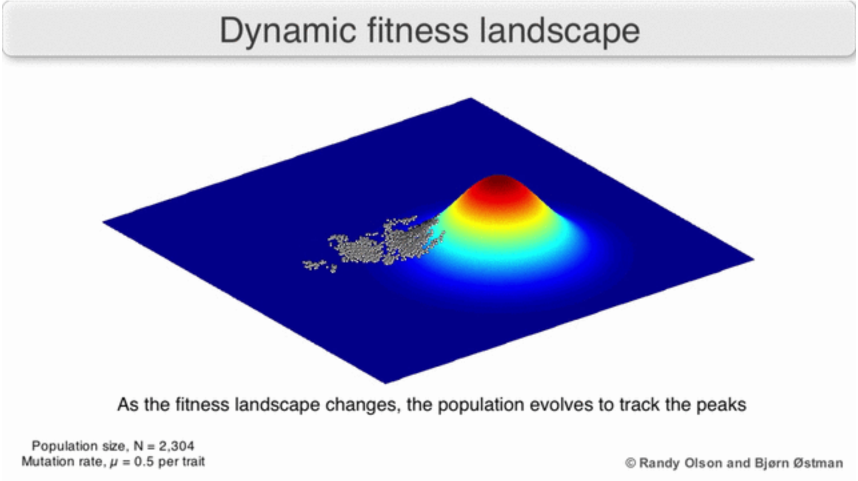
GA Numerical proof - Quod erat demonstratum



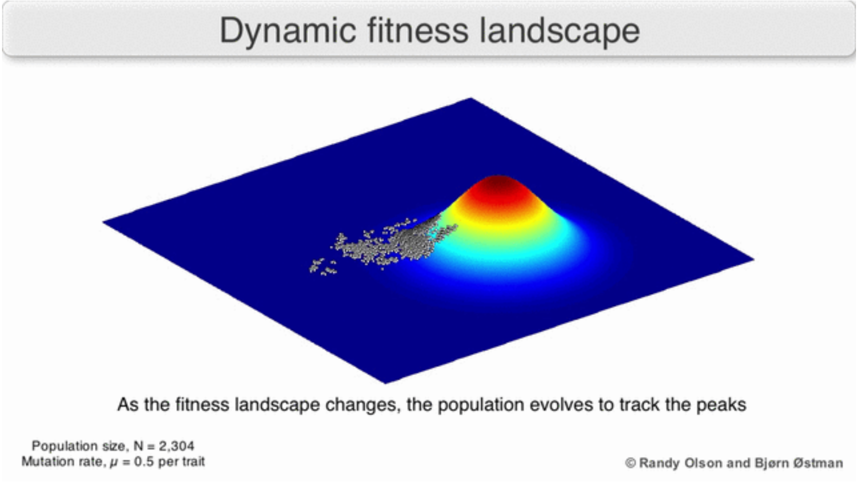
GA Numerical proof - Quod erat demonstratum



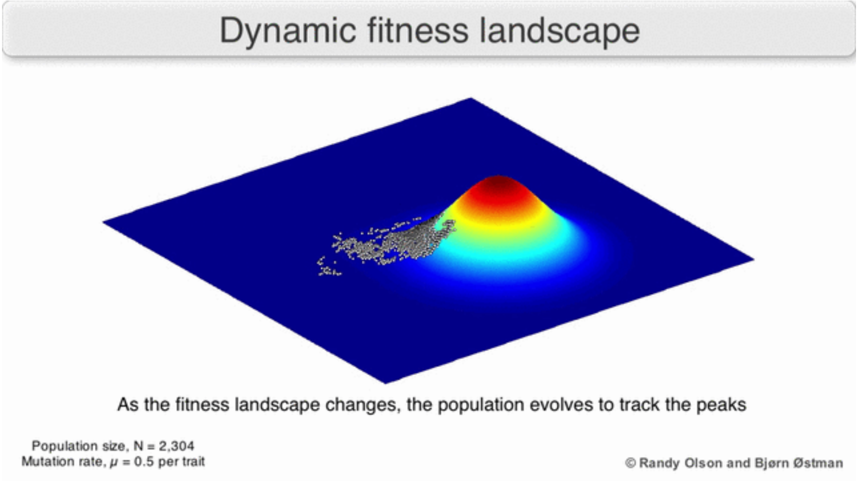
GA Numerical proof - Quod erat demonstratum



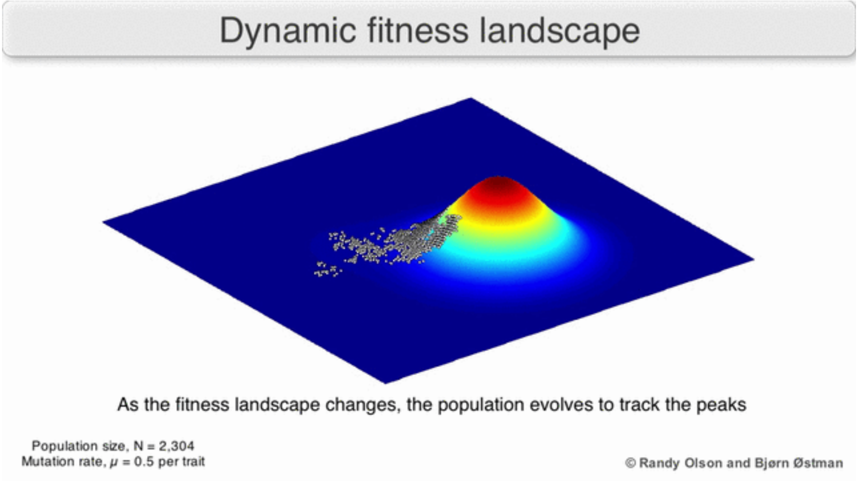
GA Numerical proof - Quod erat demonstratum



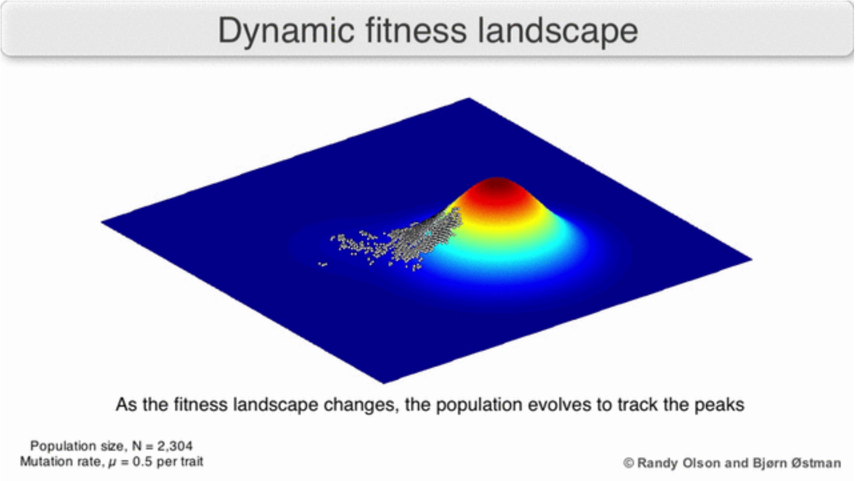
GA Numerical proof - Quod erat demonstratum



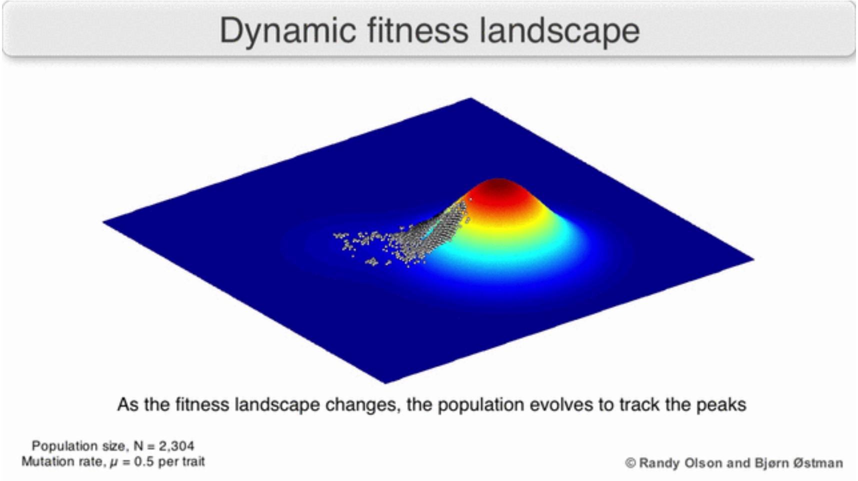
GA Numerical proof - Quod erat demonstratum



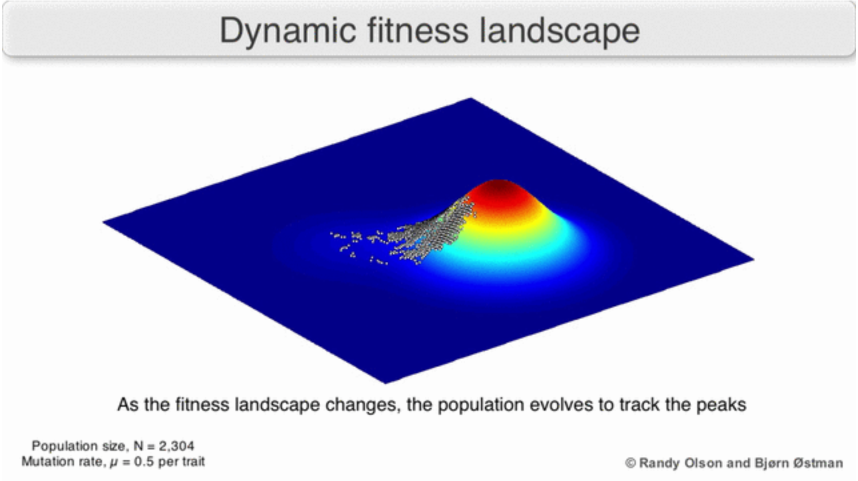
GA Numerical proof - Quod erat demonstratum



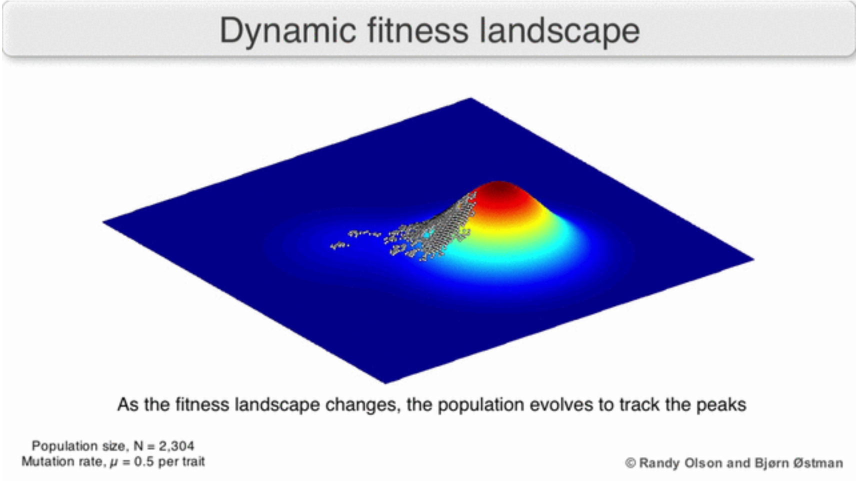
GA Numerical proof - Quod erat demonstratum



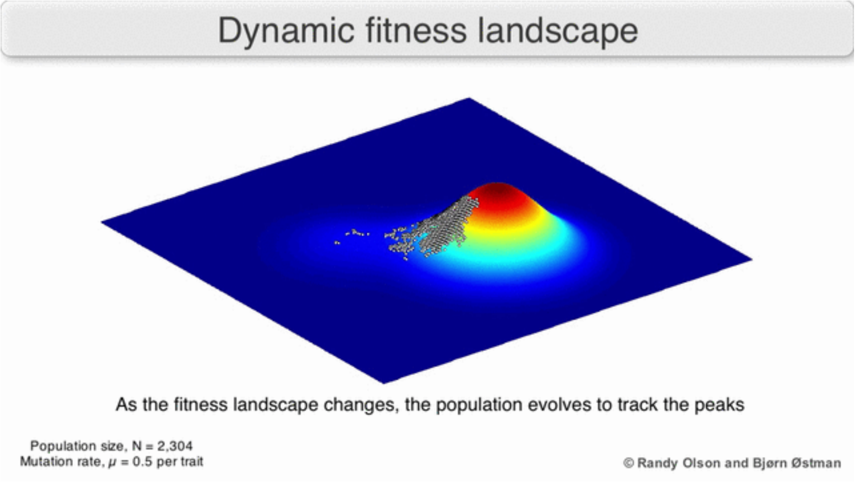
GA Numerical proof - Quod erat demonstratum



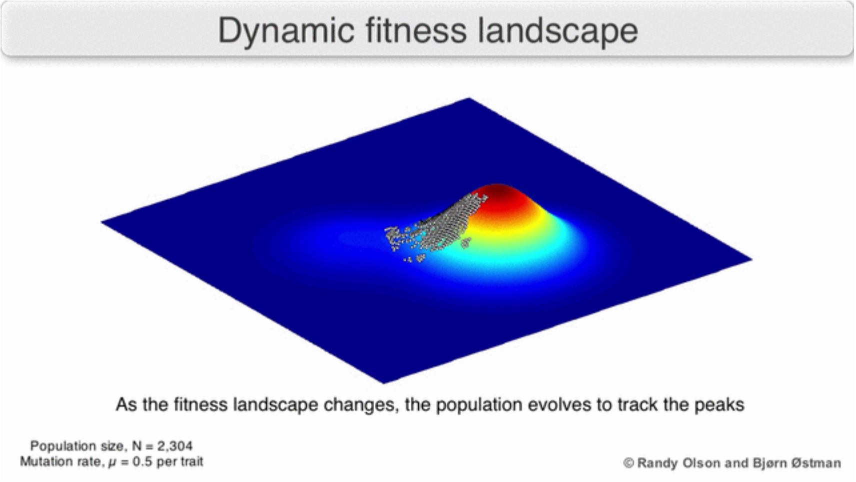
GA Numerical proof - Quod erat demonstratum



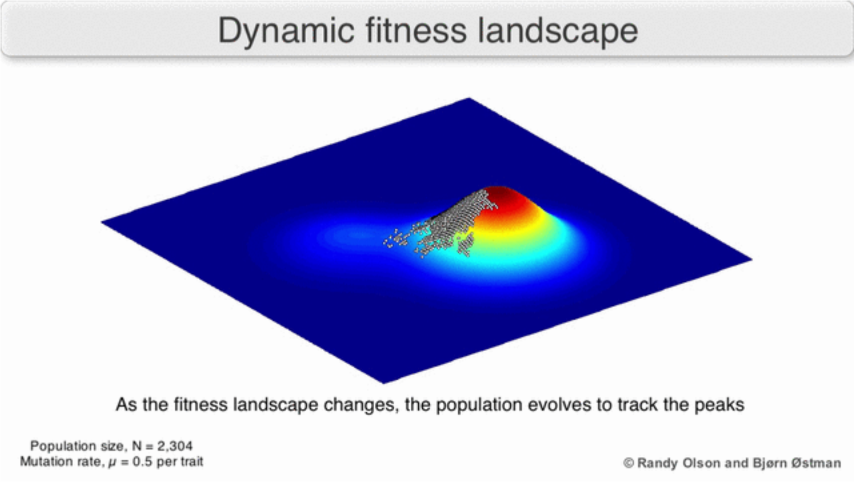
GA Numerical proof - Quod erat demonstratum



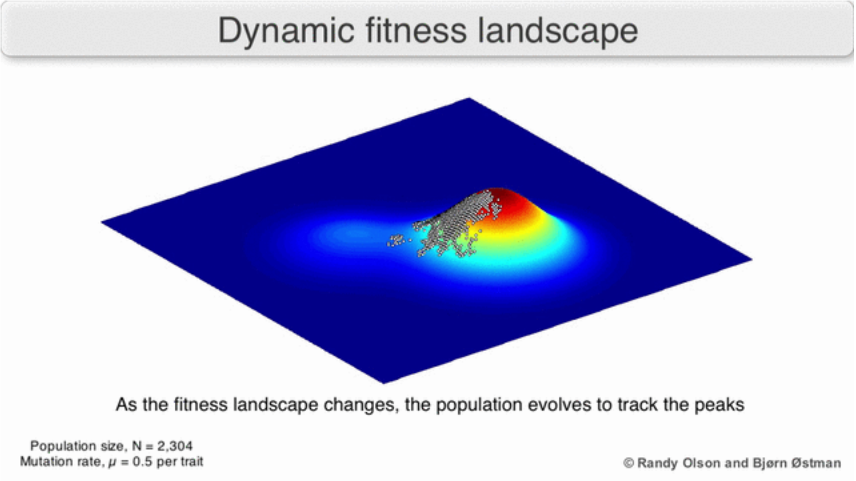
GA Numerical proof - Quod erat demonstratum



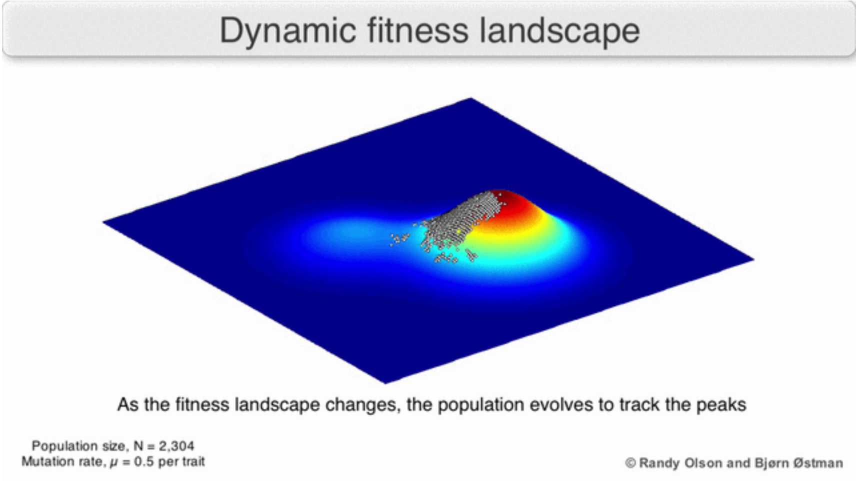
GA Numerical proof - Quod erat demonstratum



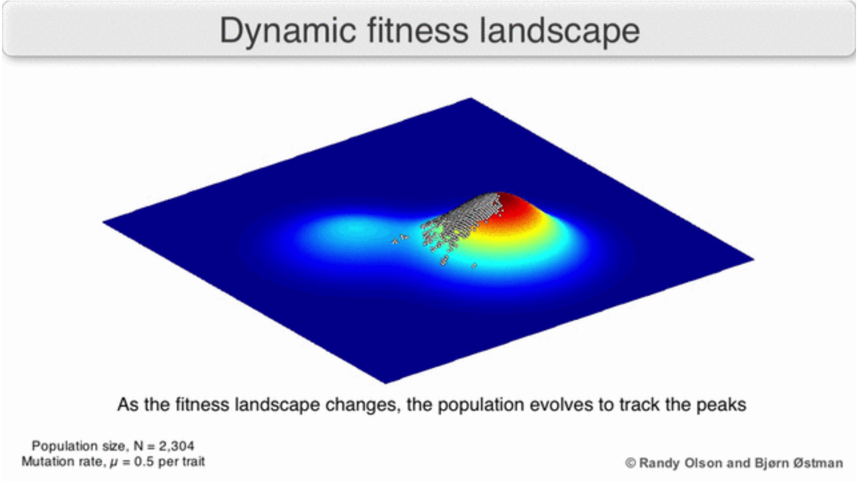
GA Numerical proof - Quod erat demonstratum



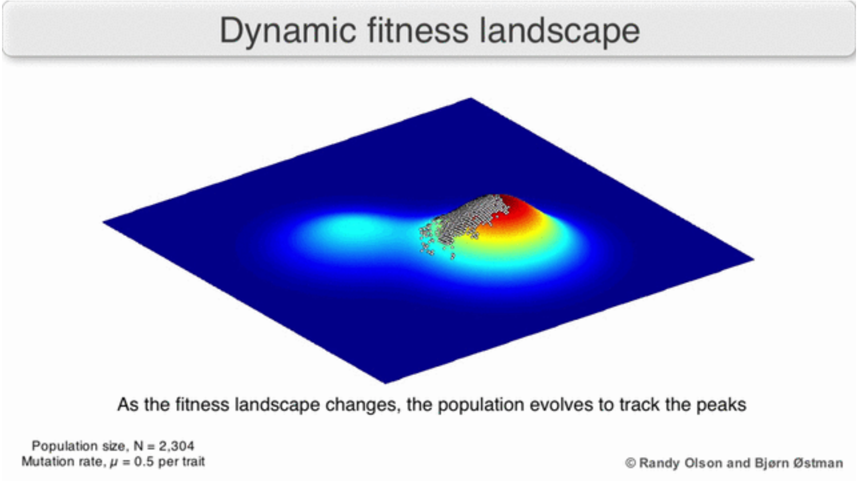
GA Numerical proof - Quod erat demonstratum



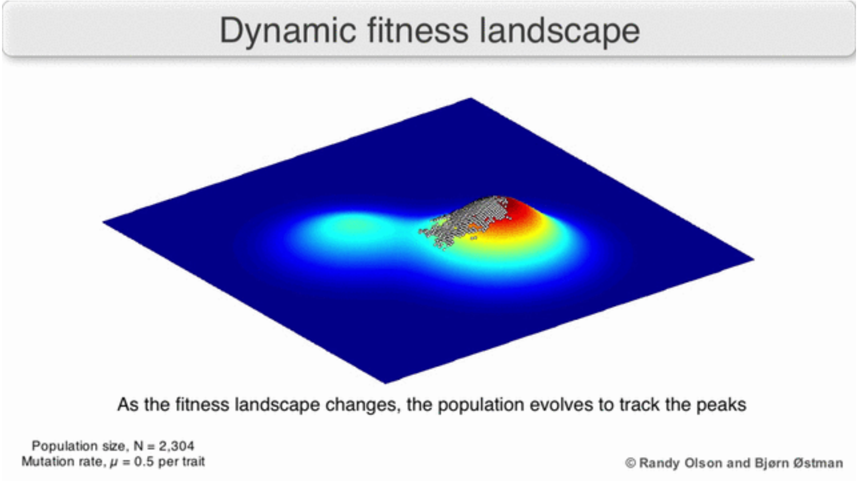
GA Numerical proof - Quod erat demonstratum



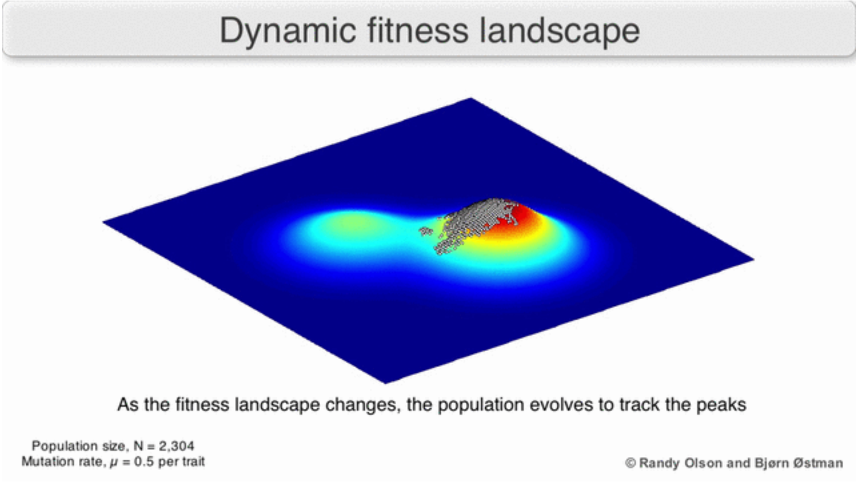
GA Numerical proof - Quod erat demonstratum



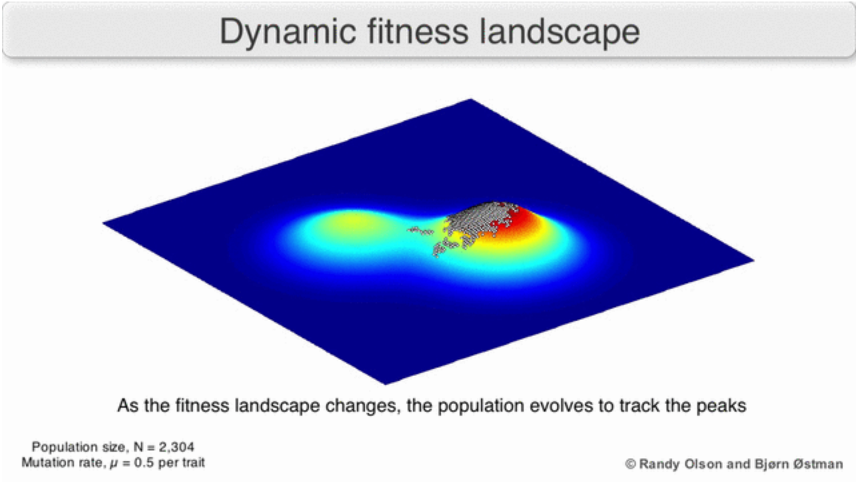
GA Numerical proof - Quod erat demonstratum



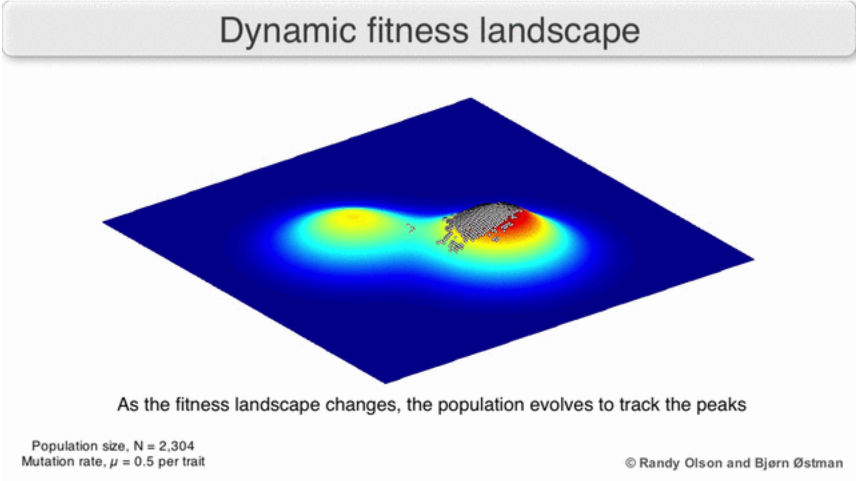
GA Numerical proof - Quod erat demonstratum



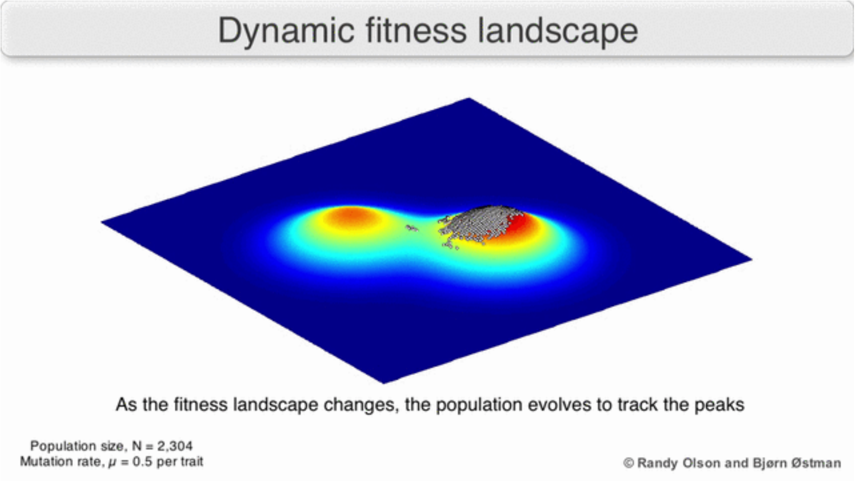
GA Numerical proof - Quod erat demonstratum



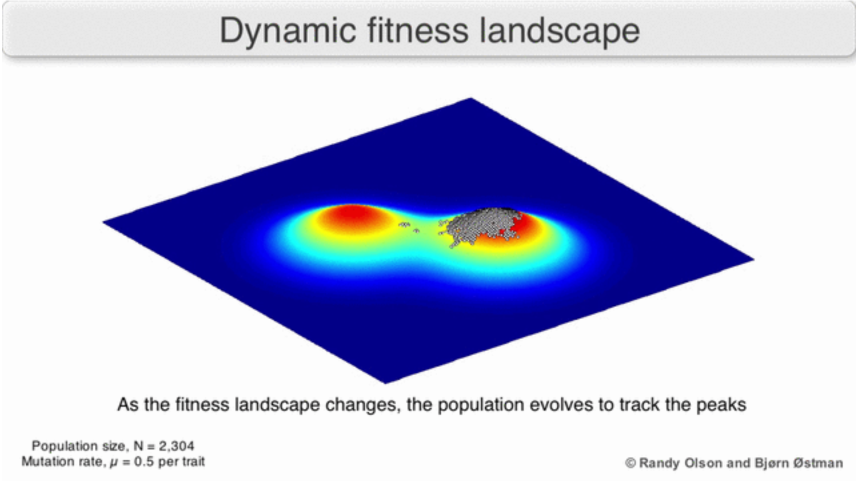
GA Numerical proof - Quod erat demonstratum



GA Numerical proof - Quod erat demonstratum



GA Numerical proof - Quod erat demonstratum



Another ex.: **Event-subset models**

$$\pi_A(dx) := \frac{1}{\mathcal{Z}_A} 1_A(x) \underbrace{\nu(dx)}_{=\text{Law}(X)=\text{example}=\mathcal{N}(0,1)}$$

$$\Downarrow$$

(To simplify) Choose a reversible local exploration of the solution space:

$\Longleftrightarrow$ A way of exploring locally $x \rightsquigarrow y$ the solution space s.t.

$$\nu(dx) \times K(x, dy) = \nu(dy) \times K(y, dx)$$

Another ex.: **Event-subset models**

$$\pi_A(dx) := \frac{1}{\mathcal{Z}_A}\, 1_A(x)\, \underbrace{\nu(dx)}_{=\text{Law}(X)=\text{example}=\mathcal{N}(0,1)}$$

$$\Downarrow$$

(To simplify) Choose a reversible local exploration of the solution space:

$\Longleftrightarrow$ A way of exploring locally $x \rightsquigarrow y$ the solution space s.t.

$$\nu(dx) \times K(x, dy) = \nu(dy) \times K(y, dx)$$

Important example $\lambda(x) \propto e^{-x^2/2}$

$$x \rightsquigarrow y = \sqrt{1-\epsilon}\; x + \sqrt{\epsilon}\; W \quad \text{with an } \mathcal{N}(0,1)\text{-sample } W$$

Design of "π_A-shakers"

- For fixed subset $A \rightsquigarrow$ **2 steps random search algorithm**

$$x(\in A) \overset{K}{\rightsquigarrow} y \rightsquigarrow z = \begin{cases} y \text{ if } y \in A \\ x \text{ otherwise} \end{cases}$$

- SA-style = Decrease gradually $A_n \downarrow$

Design of "π_A-shakers"

- For fixed subset $A \rightsquigarrow$ **2 steps random search algorithm**

$$x(\in A) \overset{K}{\rightsquigarrow} y \rightsquigarrow z = \begin{cases} y \text{ if } y \in A \\ x \text{ otherwise} \end{cases}$$

- SA-style = Decrease gradually $A_n \downarrow$

$\rightsquigarrow$ **Ex.: Uncertainty propagations in numerical codes:**

Law (**Inputs** X | **Outputs** $Y = F(X)$ $\in$ **Reference or Critical event**)

Design of "π_A-shakers"

- For fixed subset $A \rightsquigarrow$ **2 steps random search algorithm**

$$x(\in A) \overset{K}{\rightsquigarrow} y \rightsquigarrow z = \begin{cases} y \text{ if } y \in A \\ x \text{ otherwise} \end{cases}$$

- SA-style = Decrease gradually $A_n \downarrow$

$\rightsquigarrow$ **Ex.: Uncertainty propagations in numerical codes:**

$$\textbf{Law}\,(\textbf{Inputs } X \mid \textbf{Outputs } Y = F(X) \in \textbf{Reference or Critical event})$$

$$\Updownarrow$$

$$\left.\begin{array}{l}\text{Law}(X)\\ A = \{x \ : \ F(x) \in B\}\end{array}\right\} \longrightarrow \mathbb{P}(X \in A) \ \& \ \text{Law}\,(X \mid X \in A)$$

Parallel version $\rightsquigarrow$ Genetic Algorithms

N **"interacting walkers/individuals/particles/genes/. . . "**

Parallel version $\rightsquigarrow$ Genetic Algorithms

N **"interacting walkers/individuals/particles/genes/. . . "**

Note that $A_n \downarrow \Leftrightarrow$ Interpolating targets π_{A_n}

$$\pi_{A_n}(dx) \ \propto \ 1_{A_n}(x) \ \pi_{A_{n-1}}(dx)$$

Parallel version $\rightsquigarrow$ Genetic Algorithms

N **"interacting walkers/individuals/particles/genes/..."**

Note that $A_n \downarrow \Leftrightarrow$ Interpolating targets π_{A_n}

$$\pi_{A_n}(dx) \ \propto \ 1_{A_n}(x) \ \pi_{A_{n-1}}(dx)$$

and

$$\mathcal{Z}_{A_n}/\mathcal{Z}_{A_{n-1}} = \nu(1_{A_n})/\nu(1_{A_{n-1}}) = \ \pi_{A_{n-1}}(1_{A_n})$$

Parallel version $\rightsquigarrow$ Genetic Algorithms

N **"interacting walkers/individuals/particles/genes/..."**

Note that $A_n \downarrow \Leftrightarrow$ Interpolating targets π_{A_n}

$$\pi_{A_n}(dx) \; \propto \; 1_{A_n}(x) \; \pi_{A_{n-1}}(dx)$$

and

$$\mathcal{Z}_{A_n}/\mathcal{Z}_{A_{n-1}} = \nu(1_{A_n})/\nu(1_{A_{n-1}}) = \; \pi_{A_{n-1}}(1_{A_n})$$

$$\Downarrow$$

Genetic algorithm $n \rightsquigarrow (n+1)$

- **Mutation:** For a given $A_n \rightsquigarrow \perp N$ exploration with π_{A_n}-shakers.

Parallel version $\rightsquigarrow$ Genetic Algorithms

N "interacting walkers/individuals/particles/genes/..."

Note that $A_n \downarrow \Leftrightarrow$ Interpolating targets π_{A_n}

$$\pi_{A_n}(dx) \;\propto\; 1_{A_n}(x)\;\pi_{A_{n-1}}(dx)$$

and

$$\mathcal{Z}_{A_n}/\mathcal{Z}_{A_{n-1}} = \nu(1_{A_n})/\nu(1_{A_{n-1}}) = \;\pi_{A_{n-1}}(1_{A_n})$$

$$\Downarrow$$

Genetic algorithm $n \rightsquigarrow (n+1)$

- **Mutation:** For a given $A_n \rightsquigarrow \perp N$ exploration with π_{A_n}-shakers.
- **Selection:** Select the individuals that have entered into A_{n+1}.

Normalizing constants (unbiased)

$$\mathbb{P}(X \in A_n) \simeq_{N\uparrow\infty} \prod_{0\leq k<n} \underbrace{\underbrace{\text{success-proportions}}_{A_k\text{-shaked-states entering in } A_{k+1}}}_{\simeq \mathbb{P}(X\in A_{k+1} \mid X\in A_k)}$$

Normalizing constants (unbiased)

$$\mathbb{P}(X \in A_n) \simeq_{N\uparrow\infty} \prod_{0\leq k<n} \underbrace{\underbrace{\text{success-proportions}}_{A_k\text{-shaked-states entering in } A_{k+1}}}_{\simeq \mathbb{P}(X\in A_{k+1} \mid X\in A_k)}$$

Adaptive version:

Choose A_{n+1} to have a given proportion of individuals in the level.

Normalizing constants (unbiased)

$$\mathbb{P}(X \in A_n) \simeq_{N\uparrow\infty} \prod_{0\leq k<n} \underbrace{\underbrace{\text{success-proportions}}_{A_k\text{-shaked-states entering in } A_{k+1}}}_{\simeq \mathbb{P}(X\in A_{k+1} \mid X\in A_k)}$$

Adaptive version:

Choose A_{n+1} to have a given proportion of individuals in the level.

Extensions to "any" targets such as

posteriors $\pi_n(dx_n) = p(dx_n|y_0,\ldots,y_n)$ **(BAYES/FILTER),**

$p(d\theta \mid (y_0,\ldots,y_n))$ **(Hidden Markov),... later in the lectures**

Genetic Algorithms $=$GA $\subset$ meta-heuristic:

Genetic Algorithms =GA ⊂ meta-heuristic:

- "**GA** = A metaheuristic inspired by the process of natural selection "Wikipedia; "mimic the way natural selection occurs in living things." (Genius blog)

Genetic Algorithms =GA $\subset$ meta-heuristic:

- "**GA** = A metaheuristic inspired by the process of natural selection "Wikipedia; "mimic the way natural selection occurs in living things." (Genius blog)
- " **GA** = Probably the most popular evolutionary algorithms with a diverse range of applications.... A model or abstraction of biological evolution based on Charles Darwin's theory of natural selection." Scholarpedia

Genetic Algorithms =GA $\subset$ meta-heuristic:

- "**GA** = A metaheuristic inspired by the process of natural selection "Wikipedia; "mimic the way natural selection occurs in living things." (Genius blog)
- " **GA** = Probably the most popular evolutionary algorithms with a diverse range of applications.. . . A model or abstraction of biological evolution based on Charles Darwin's theory of natural selection." Scholarpedia
- "**GA** = One of the most extensively utilized meta-heuristic algorithms" (Review of metaheuristic (23))

Genetic Algorithms =GA $\subset$ meta-heuristic:

- "**GA** = A metaheuristic inspired by the process of natural selection "Wikipedia; "mimic the way natural selection occurs in living things." (Genius blog)
- " **GA** = Probably the most popular evolutionary algorithms with a diverse range of applications.... A model or abstraction of biological evolution based on Charles Darwin's theory of natural selection." Scholarpedia
- "**GA** = One of the most extensively utilized meta-heuristic algorithms" (Review of metaheuristic (23))
- "**GA** = Computer programs that evolve in ways that resemble natural selection can solve complex problems even their creators do not ully understand" (`J.H. Holland`, `Scientific American 1992`)

Some tries to understand

- "In mathematical optimization, a **metaheuristic** is a higher-level procedure or **heuristic** designed to find, generate, tune, or select **a heuristic.** " Wikipedia Metaheuristic def.
- " **Heuristic** is a technique designed for problem solving more quickly when classic methods are too slow or fail to find any exact solution in a search space." Wikipedia heuristic def.
- A common name for algorithms with randomization, or based on some magic natural/physical phenomena?

Mathematical definition:

$\rightsquigarrow$ **Heuristic = conjecture/informal/experience-based/numerical proof**

Origins of the GA meta-heuristic:

- **Taken from Alan Turing's 1950's article Computing Machinery and Intelligence** $\rightsquigarrow$ **Design learning machine**
  - "Changes of the machine = mutation".
  - "Natural selection = judgment of the experimenter (rewards increase repetition of the events which led up to it)"

Origins of the GA meta-heuristic:

- **Taken from Alan Turing's 1950's article Computing Machinery and Intelligence** $\rightsquigarrow$ **Design learning machine**
  - "Changes of the machine = mutation".
  - "Natural selection = judgment of the experimenter (rewards increase repetition of the events which led up to it)"

- **Equivalent Mutation/Selection (Monte Carlo) heuristics:**
  Pruned-enriched Rosenbluth/Grassberger 1955; `Genetic Monte Carlo systems, Fraser 1954`; Reconfiguration DMC Hetherington 1984; `GA-computer prog. Holland 1992`, `Bootstrap filter Gordon-Salmon-Smith 1993`,...

Origins of the GA meta-heuristic:

- **Taken from Alan Turing's 1950's article Computing Machinery and Intelligence** $\rightsquigarrow$ **Design learning machine**
  - "Changes of the machine = mutation".
  - "Natural selection = judgment of the experimenter (rewards increase repetition of the events which led up to it)"

- **Equivalent Mutation/Selection (Monte Carlo) heuristics:**
  Pruned-enriched Rosenbluth/Grassberger 1955; `Genetic Monte Carlo systems, Fraser 1954`; Reconfiguration DMC Hetherington 1984; `GA-computer prog. Holland 1992`, `Bootstrap filter Gordon-Salmon-Smith 1993`,...

First/Vintage rigorous proofs

Mean field Interacting particle samplers+Non asymptotic estimates + unbiasedness unnormalized models + terminology "particle filters"

$\rightsquigarrow$ **(MPRF-1996)**

Conditional mutations, branching/mean field interacting particle models
$\rightsquigarrow$ (MPRF 1998), (AAP-1998), . . .

Continuous time/discretisations/LDP/CLT/Expo. Concentration/Empirical Processes,. . .

LDP (SPA1998), Kushner Stratonovitch eq. (Ad-AP-1999), CLT (AAP1999), FK (SPA2000), Emp. proc. (JTP2000), BIPS (Sp2000), . . .

With many co-authors (cf. `(website)`): A. Guionnet, D. Crisan, T. Lyons, L. Miclo, J. Jacod, P. Protter, A. Doucet, A. Jasra, M. Ledoux, J. Garnier, D. Dawson, L. Murray, F. Caron, E. Rio, L. Wu, F. Patras, E. Moulines, M. Arnaudon, S.S. Singh,. . .

CAUTION: N = precision (nb of samples/particles/walkers)

Maths literature (⊃ some of the previous articles) abounds with fancy bounds/theo of the type:

"Theorem": Mean error/bias/variance/estimate at time $t \leq e^{7t}/N$

CAUTION: N = precision (nb of samples/particles/walkers)

Maths literature (⊃ some of the previous articles) abounds with fancy bounds/theo of the type:

"*Theorem*": Mean error/bias/variance/estimate at time $t \leq e^{7t}/N$

or of type $e^{c_1 t}/N$, $c_2(t)/N$ or sometimes $\forall t\ \exists c$ s.t....

CAUTION: N = precision (nb of samples/particles/walkers)

Maths literature ($\supset$ some of the previous articles) abounds with fancy bounds/theo of the type:

"Theorem": Mean error/bias/variance/estimate at time $t \leq e^{7t}/N$

or of type $e^{c_1 t}/N$, $c_2(t)/N$ or sometimes $\forall t\ \exists c$ s.t....

BUT $t \geq 27 \Longrightarrow e^{7t} > 10^{82}$ Nb atoms $\in$ visible universe

CAUTION: N = precision (nb of samples/particles/walkers)

Maths literature ($\supset$ some of the previous articles) abounds with fancy bounds/theo of the type:

"Theorem": Mean error/bias/variance/estimate at time $t \leq e^{7t}/N$

or of type $e^{c_1 t}/N$, $c_2(t)/N$ or sometimes $\forall t\ \exists c$ s.t....

BUT $t \geq 27 \Longrightarrow e^{7t} > 10^{82}$ Nb atoms $\in$ visible universe

CAUTION: N = precision (nb of samples/particles/walkers)

Maths literature ($\supset$ some of the previous articles) abounds with fancy bounds/theo of the type:

"Theorem": Mean error/bias/variance/estimate at time $t \leq e^{7t}/N$

or of type $e^{c_1 t}/N$, $c_2(t)/N$ or sometimes $\forall t\ \exists c$ s.t....

BUT $t \geq 27 \Longrightarrow e^{7t} > 10^{82}$ Nb atoms $\in$ visible universe

Impossible to run/sample/evolve

CAUTION: N = precision (nb of samples/particles/walkers)

Maths literature ($\supset$ some of the previous articles) abounds with fancy bounds/theo of the type:

"Theorem": Mean error/bias/variance/estimate at time $t \leq e^{7t}/N$

or of type $e^{c_1 t}/N$, $c_2(t)/N$ or sometimes $\forall t\ \exists c$ s.t....

BUT $t \geq 27 \Longrightarrow e^{7t} > 10^{82}$ Nb atoms $\in$ visible universe

Impossible to run/sample/evolve OR <u>inadequate derivation !</u>

ONE KEY IDEA: **Stochastic perturbation analysis**

Origins/First time-uniform estimates for mean field particle samplers

$\rightsquigarrow$ (CRAS1999)/(IHP2021),(SP2000),...

$\rightsquigarrow$ cf. books+refs: (FK2004), (FTML2012), (CRC2013)

Extensions to diffusion flows:

Riccati matrix flows (IHP2020)/(Arxiv-2017)

McKean-Vlasov diffusions (AAP2020)/(Arxiv-2019),

SDEs interpolations $\rightsquigarrow$ (CRAS2020)/(SPA2022)/(Arxiv-2019),...

Lecture intro/contents

Probabilistic modeling, GA Monte Carlo methodologies/toolbox

$\oplus$

Operator theory/Stochastic analysis tools (stability+perturbation theory)

Many open pb

Most time-uniform estimates = Compact state space conditions

- To control first/second order operators in interpolation formulae.
- Bounded observable/test functions
- Ok pour SDE on compact sets (reflection on boundaries) but not linear/Gaussian SDE.

Many open pb

For instance in the context of Linear/Gaussian models:

"50y of numerical proofs + justification by comparisons"

(DMC, particle filters) $\simeq$ (quantum harmonic oscillators/Kalman filters)
[$\exists$ some rigorous `CLT (Fields2002)`, `+ (book chapter 2001)` but...]

Many open pb

For instance in the context of Linear/Gaussian models:

"50y of numerical proofs + justification by comparisons"

(DMC, particle filters) $\simeq$ (quantum harmonic oscillators/Kalman filters)
[$\exists$ some rigorous `CLT (Fields2002)`, `+ (book chapter 2001)` but...]

$\rightsquigarrow$ **CAUTION:**

• *CLT/Asymptotic variance uniformly bounded in time* `Whiteley (AAP-2013)`

Many open pb

For instance in the context of Linear/Gaussian models:

"50y of numerical proofs + justification by comparisons"

(DMC, particle filters) $\simeq$ (quantum harmonic oscillators/Kalman filters)
[$\exists$ some rigorous CLT (Fields2002), + (book chapter 2001) but. . .]

$\rightsquigarrow$ **CAUTION:**

• *CLT/Asymptotic variance uniformly bounded in time* Whiteley (AAP-2013)**BUT** *non asymptotic variance may* **blows up/diverge** *(for unstable mutations). (MCSS-2023)*

Many open pb

For instance in the context of Linear/Gaussian models:

"50y of numerical proofs + justification by comparisons"

(DMC, particle filters) $\simeq$ (quantum harmonic oscillators/Kalman filters)
[$\exists$ some rigorous `CLT (Fields2002)`, **+** `(book chapter 2001)` but. . .]

$\rightsquigarrow$ **CAUTION:**

- *CLT/Asymptotic variance uniformly bounded in time* `Whiteley (AAP-2013)` **BUT** *non asymptotic variance may* **blows up/diverge** *(for unstable mutations). (MCSS-2023)*

- **After 50y $\rightsquigarrow$ First non asymptotic time-uniform estimates applying to linear/gaussian** *models [but not to all] (Arxiv 2024)*.

Many open pb

For instance in the context of Linear/Gaussian models:

"50y of numerical proofs + justification by comparisons"

(DMC, particle filters) $\simeq$ (quantum harmonic oscillators/Kalman filters)
[$\exists$ some rigorous `CLT (Fields2002)`, `+ (book chapter 2001)` but...]

$\rightsquigarrow$ **CAUTION:**

• *CLT/Asymptotic variance uniformly bounded in time* `Whiteley (AAP-2013)` **BUT** *non asymptotic variance may* **blows up/diverge** *(for unstable mutations). (MCSS-2023)*

• **After 50y $\rightsquigarrow$ First non asymptotic time-uniform estimates applying to linear/gaussian** *models [but not to all] (Arxiv 2024).*

⇝ Website **GA** Monte Carlo methodologies

⇝ Website **GA** Monte Carlo methodologies

A single mathematical framework for a variety of equivalent algorithms

⇝ Website **GA** Monte Carlo methodologies

A single mathematical framework for a variety of equivalent algorithms

- **Theoretical physics:** Pruned-Enriched Rosenbluth Method (a.k.a. PERM - molecular simulation), Reconfiguration/Stochastic Reconfiguration, Population Monte Carlo method, Diffusion Quantum Monte Carlo.

⇝ Website **GA** Monte Carlo methodologies

A single mathematical framework for a variety of equivalent algorithms

- **Theoretical physics:** Pruned-Enriched Rosenbluth Method (a.k.a. PERM - molecular simulation), Reconfiguration/Stochastic Reconfiguration, Population Monte Carlo method, Diffusion Quantum Monte Carlo.
- **Math Biology:** Branching/Elimination, Killing/Regeneration, Dying/Branching, Killing/Cloning, Fleming-Viot & Moran, ...

⇝ Website **GA** Monte Carlo methodologies

A single mathematical framework for a variety of equivalent algorithms

- **Theoretical physics:** Pruned-Enriched Rosenbluth Method (a.k.a. PERM - molecular simulation), Reconfiguration/Stochastic Reconfiguration, Population Monte Carlo method, Diffusion Quantum Monte Carlo.
- **Math Biology:** Branching/Elimination, Killing/Regeneration, Dying/Branching, Killing/Cloning, Fleming-Viot & Moran, ...
- **Signal processing:** Particle filter, convolution particle filters, the Bootstrap filter and the Monte Carlo filter, Interacting Kalman filter methodology, a.k.a. Rao-Blackwellized particle filter.

$\rightsquigarrow$ Website **GA** Monte Carlo methodologies

A single mathematical framework for a variety of equivalent algorithms

- **Theoretical physics:** Pruned-Enriched Rosenbluth Method (a.k.a. PERM - molecular simulation), Reconfiguration/Stochastic Reconfiguration, Population Monte Carlo method, Diffusion Quantum Monte Carlo.
- **Math Biology:** Branching/Elimination, Killing/Regeneration, Dying/Branching, Killing/Cloning, Fleming-Viot & Moran, ...
- **Signal processing:** Particle filter, convolution particle filters, the Bootstrap filter and the Monte Carlo filter, Interacting Kalman filter methodology, a.k.a. Rao-Blackwellized particle filter.
- **Probability and (Bayesian) Statistics:** Sequential Monte Carlo, adaptive multilevel splitting, adaptive SMC, Markovian anticipation, Pilot explorations. Interacting Metropolis algorithm/Simulated algorithms.

⇝ Website **GA** Monte Carlo methodologies

A single mathematical framework for a variety of equivalent algorithms

- **Theoretical physics:** Pruned-Enriched Rosenbluth Method (a.k.a. PERM - molecular simulation), Reconfiguration/Stochastic Reconfiguration, Population Monte Carlo method, Diffusion Quantum Monte Carlo.
- **Math Biology:** Branching/Elimination, Killing/Regeneration, Dying/Branching, Killing/Cloning, Fleming-Viot & Moran, ...
- **Signal processing:** Particle filter, convolution particle filters, the Bootstrap filter and the Monte Carlo filter, Interacting Kalman filter methodology, a.k.a. Rao-Blackwellized particle filter.
- **Probability and (Bayesian) Statistics:** Sequential Monte Carlo, adaptive multilevel splitting, adaptive SMC, Markovian anticipation, Pilot explorations. Interacting Metropolis algorithm/Simulated algorithms.
- **Operation research:** GA, go-with-the-winner, Gibbs-cloning, subset simulation.

⇝ Website GA Monte Carlo methodologies

A single mathematical framework for a variety of equivalent algorithms

- **Theoretical physics:** Pruned-Enriched Rosenbluth Method (a.k.a. PERM - molecular simulation), Reconfiguration/Stochastic Reconfiguration, Population Monte Carlo method, Diffusion Quantum Monte Carlo.
- **Math Biology:** Branching/Elimination, Killing/Regeneration, Dying/Branching, Killing/Cloning, Fleming-Viot & Moran, ...
- **Signal processing:** Particle filter, convolution particle filters, the Bootstrap filter and the Monte Carlo filter, Interacting Kalman filter methodology, a.k.a. Rao-Blackwellized particle filter.
- **Probability and (Bayesian) Statistics:** Sequential Monte Carlo, adaptive multilevel splitting, adaptive SMC, Markovian anticipation, Pilot explorations. Interacting Metropolis algorithm/Simulated algorithms.
- **Operation research:** GA, go-with-the-winner, Gibbs-cloning, subset simulation.

Meta-heuristic ⇝ Meta-Theorems ⊕ Many open problems . . .

Ex.: application's dependent+more refined analysis

⇝ RW $\in$ Tube (SAA-2018), Coupled Harmonic Osc. (CMM-2023),. . .

Some concrete applications of GA Monte Carlo methods

⇝ `List of real world applications of GA (wikipedia)`

⊕

⇝ (GA & FK website links)

(ALEA/CQFD/ASTRAL INRIA team(s)) `(website industry transfers)`

- Watermarking - ARC INRIA RARE (2009 - 7 INRIA teams project).
- Satellite debris tracking/control (2016): ONERA & CNES.
- Nuclear plant security (2011): EDF R&D Chatou.
- Offshore structures reliability (2009-2010): IFREMER.
- Sparse antenna arrays design (2009): CEA-CESTA.
- Signal deconvolution (2010-2012): CEA CESTA.
- Financial options (2010-2012): EDF R&D Clamart.
- Interacting Kalman: Dassault Aviation (2011 with INRIA-EPI I4S) .
- RADAR/SONAR/GPS: LAAS-CNRS/STCAN (1990-1994),

Since 2007 (INRIA/DCNS/Naval Group) $\rightsquigarrow$

Multi-objects/targets tracking, navigation, multi-sensors, . . .

INRIA team projet major partnership 2020 $=$ `ASTRAL` $\oplus$ `Naval Group`

$\rightsquigarrow$ `INRIA Research (only) positions website (2024 competition ended` $\rightsquigarrow$ `2025 starts "November 2024.")`

Markov chains (finite spaces)

Markov chain = "sequence of r.v."

$$X_0 \rightsquigarrow X_1 \rightsquigarrow \ldots \rightsquigarrow X_{n-1} \rightsquigarrow X_n$$

Markov chains (finite spaces)

Markov chain = "sequence of r.v."

$$X_0 \rightsquigarrow X_1 \rightsquigarrow \ldots \rightsquigarrow X_{n-1} \rightsquigarrow X_n$$

Example: $E = \{1, \ldots, d\}$

$$\underbrace{\mathbb{P}(X_n = j)}_{=\eta_n(j)} = \sum_{1 \le i \le d} \underbrace{\mathbb{P}(X_{n-1} = i)}_{=\eta_{n-1}(i)} \; \underbrace{\mathbb{P}(X_n = j \mid X_{n-1} = i)}_{=P_n(i,j)}$$

Markov chains (finite spaces)

Markov chain = "sequence of r.v."

$$X_0 \rightsquigarrow X_1 \rightsquigarrow \ldots \rightsquigarrow X_{n-1} \rightsquigarrow X_n$$

Example: $E = \{1, \ldots, d\}$

$$\underbrace{\mathbb{P}(X_n = j)}_{=\eta_n(j)} = \sum_{1 \le i \le d} \underbrace{\mathbb{P}(X_{n-1} = i)}_{=\eta_{n-1}(i)} \; \underbrace{\mathbb{P}(X_n = j \mid X_{n-1} = i)}_{=P_n(i,j)}$$

$$\Updownarrow$$

Matrix synthetic notation:

$$\eta_n = \eta_{n-1} P_n = \ldots = \eta_0 P_1 P_2 \ldots P_n$$

Markov chains (general state spaces)

Discrete time dynamical system

$$X_n = F_n(X_{n-1}, W_n) \qquad \text{with } \perp \text{ random variables } W_n$$

Markov chains (general state spaces)

Discrete time dynamical system

$$X_n = F_n(X_{n-1}, W_n) \qquad \text{with } \perp \text{ random variables } W_n$$

Example

$$X_n = \sqrt{1-\epsilon}\; X_{n-1} + \sqrt{\epsilon}\; W_n \qquad \text{with } \perp \text{ iid } W_n \sim N(0,1)$$

Markov chains (general state spaces)

Discrete time dynamical system

$$X_n = F_n(X_{n-1}, W_n) \qquad \text{with } \perp \text{ random variables } W_n$$

Example

$$X_n = \sqrt{1-\epsilon}\; X_{n-1} + \sqrt{\epsilon}\; W_n \qquad \text{with } \perp \text{ iid } W_n \sim N(0,1)$$

or in terms of Markov transitions on some state spaces E_n:

$$\mathbb{P}(X_n \in dx_n \mid X_{n-1} = x_{n-1}) = P_n(x_{n-1}, dx_n)$$

Markov chains (general state spaces)

Discrete time dynamical system

$$X_n = F_n(X_{n-1}, W_n) \qquad \text{with } \perp \text{ random variables } W_n$$

Example

$$X_n = \sqrt{1-\epsilon}\; X_{n-1} + \sqrt{\epsilon}\; W_n \qquad \text{with } \perp \text{ iid } W_n \sim N(0,1)$$

or in terms of Markov transitions on some state spaces E_n:

$$\mathbb{P}(X_n \in dx_n \mid X_{n-1} = x_{n-1}) = P_n(x_{n-1}, dx_n)$$

Example - historical process

$$X_n = (X'_0, X'_1, \ldots, X'_n)$$

Markov chains (general state spaces $X_n \in E_n$)

Evolution equations:

$$\underbrace{\mathbb{P}(X_n \in dx_n)}_{=\eta_n(dx_n)} = \int_{E_{n-1}} \underbrace{\mathbb{P}(X_{n-1} \in dx_{n-1})}_{=\eta_{n-1}(dx_{n-1})} \underbrace{\mathbb{P}(X_n \in dx_n \mid X_{n-1} = x_{n-1})}_{=P_n(x_{n-1},dx_n)}$$

Markov chains (general state spaces $X_n \in E_n$)

Evolution equations:

$$\underbrace{\mathbb{P}(X_n \in dx_n)}_{=\eta_n(dx_n)} = \int_{E_{n-1}} \underbrace{\mathbb{P}(X_{n-1} \in dx_{n-1})}_{=\eta_{n-1}(dx_{n-1})} \underbrace{\mathbb{P}(X_n \in dx_n \mid X_{n-1} = x_{n-1})}_{=P_n(x_{n-1}, dx_n)}$$

$$\Updownarrow$$

(Integral) operator synthetic notation:

$$\eta_n = \eta_{n-1} P_n = \ldots = \eta_0 P_1 P_2 \ldots P_n$$

Markov chains (general state spaces $X_n \in E_n$)

Evolution equations:

$$\underbrace{\mathbb{P}(X_n \in dx_n)}_{=\eta_n(dx_n)} = \int_{E_{n-1}} \underbrace{\mathbb{P}(X_{n-1} \in dx_{n-1})}_{=\eta_{n-1}(dx_{n-1})} \underbrace{\mathbb{P}(X_n \in dx_n \mid X_{n-1} = x_{n-1})}_{=P_n(x_{n-1},dx_n)}$$

$$\Updownarrow$$

(Integral) operator synthetic notation:

$$\eta_n = \eta_{n-1} P_n = \ldots = \eta_0 P_1 P_2 \ldots P_n$$

Semigroup notation (for $k \leq l \leq n$)

$$P_{k,n} := P_{k+1} P_{k+2} \ldots P_n$$

Markov chains (general state spaces $X_n \in E_n$)

Evolution equations:

$$\underbrace{\mathbb{P}(X_n \in dx_n)}_{=\eta_n(dx_n)} = \int_{E_{n-1}} \underbrace{\mathbb{P}(X_{n-1} \in dx_{n-1})}_{=\eta_{n-1}(dx_{n-1})} \underbrace{\mathbb{P}(X_n \in dx_n \mid X_{n-1} = x_{n-1})}_{=P_n(x_{n-1},dx_n)}$$

$$\Updownarrow$$

(Integral) operator synthetic notation:

$$\eta_n = \eta_{n-1} P_n = \ldots = \eta_0 P_1 P_2 \ldots P_n$$

Semigroup notation (for $k \leq l \leq n$)

$$P_{k,n} := P_{k+1} P_{k+2} \ldots P_n = P_{k,l} P_{l,n} \quad \text{and} \quad P_{n,n} = Id \quad \Longrightarrow \quad \eta_l P_{l,n} = \eta_n$$

Linear integral equations

Solving linear measure valued equations:

$$\eta_n = \Phi_n(\eta_{n-1}) := \eta_{n-1} P_n = \mathrm{Law}(X_n)$$

with a (non unique) Markov chain interpretation

$$X_n \sim P_n(X_{n-1}, dx_n)$$

Linear integral equations

Solving linear measure valued equations:

$$\eta_n = \Phi_n(\eta_{n-1}) := \eta_{n-1}P_n = \mathrm{Law}(X_n)$$

with a (non unique) Markov chain interpretation

$$X_n \sim P_n(X_{n-1}, dx_n)$$

$\rightsquigarrow$ **N independent copies $(X_n^i)_{1\leq i\leq N}$ of the chain X_n:**

$$\frac{1}{N}\sum_{1\leq i\leq n} f(X_n^i) = \int f(x_n)\ \underbrace{\frac{1}{N}\sum_{1\leq i\leq N}\delta_{X_n^i}(dx_n)}_{:=\eta_n^{\mathbf{N}}(dx_n)} \simeq_{N\uparrow\infty} \int f(x_n)\ \eta_n(dx_n)$$

Linear integral equations

Solving linear measure valued equations:

$$\eta_n = \Phi_n(\eta_{n-1}) := \eta_{n-1}P_n = \mathrm{Law}(X_n)$$

with a (non unique) Markov chain interpretation

$$X_n \sim P_n(X_{n-1}, dx_n)$$

$\rightsquigarrow$ **N independent copies $(X_n^i)_{1\leq i\leq N}$ of the chain X_n:**

$$\frac{1}{N}\sum_{1\leq i\leq n} f(X_n^i) = \int f(x_n)\ \underbrace{\frac{1}{N}\sum_{1\leq i\leq N}\delta_{X_n^i}(dx_n)}_{:=\eta_n^{\mathbf{N}}(dx_n)} \simeq_{N\uparrow\infty} \int f(x_n)\ \eta_n(dx_n)$$

Equivalent formulation:

$$\eta_n^{\mathbf{N}}(f) \simeq_{N\uparrow\infty} \eta_n(f)$$

Linear integral equations

Solving linear measure valued equations:

$$\eta_n = \Phi_n(\eta_{n-1}) := \eta_{n-1}P_n = \mathrm{Law}(X_n)$$

with a (non unique) Markov chain interpretation

$$X_n \sim P_n(X_{n-1}, dx_n)$$

$\rightsquigarrow$ N independent copies $(X_n^i)_{1\leq i\leq N}$ of the chain X_n:

$$\frac{1}{N}\sum_{1\leq i\leq n} f(X_n^i) = \int f(x_n)\, \underbrace{\frac{1}{N}\sum_{1\leq i\leq N} \delta_{X_n^i}(dx_n)}_{:=\eta_n^{\mathbf{N}}(dx_n)} \simeq_{N\uparrow\infty} \int f(x_n)\, \eta_n(dx_n)$$

Equivalent formulation:

$$\eta_n^{\mathbf{N}}(f) \simeq_{N\uparrow\infty} \eta_n(f) \rightsquigarrow \textit{more compact form} \quad \eta_n^{\mathbf{N}} \simeq_{N\uparrow\infty} \eta_n$$

Perturbation formulation

$$\mathbb{V}_n^N(f) := \frac{1}{\sqrt{N}} \sum_{1\leq i\leq N} \left(f(X_n^i) - \mathbb{E}\left(f(X_n^i) \mid X_{n-1}^i\right)\right)$$

$$\Downarrow$$

Note $\mathbb{V}_n^N(f)$ **is centered and when** $\mathrm{osc}(f) \leq 1$ **we have**

$$\mathbb{E}\left(\left[\mathbb{V}_n^N(f)\right]^2 \mid (X_{n-1}^i)_i\right) = \int \eta_{n-1}^N(dx)\ P_n((f - P_n(f)(x))^2)(x) \leq 1$$

Perturbation formulation

$$\mathbb{V}_n^N(f) := \frac{1}{\sqrt{N}} \sum_{1\leq i\leq N} \left(f(X_n^i) - \mathbb{E}\left(f(X_n^i) \mid X_{n-1}^i \right) \right)$$

$$\Downarrow$$

Note $\mathbb{V}_n^N(f)$ is centered and when $\operatorname{osc}(f) \leq 1$ we have

$$\mathbb{E}\left(\left[\mathbb{V}_n^N(f)\right]^2 \mid (X_{n-1}^i)_i \right) = \int \eta_{n-1}^N(dx)\ P_n((f - P_n(f)(x))^2)(x) \leq 1$$

More compact perturbation formulas

$$\eta_n^{\mathbf{N}} = \eta_{n-1}^{\mathbf{N}} P_n + \frac{\mathbf{1}}{\sqrt{\mathbf{N}}}\ \mathbb{V}_n^N$$

Perturbation formulation

$$\mathbb{V}_n^N(f) := \frac{1}{\sqrt{N}} \sum_{1\leq i\leq N} \left(f(X_n^i) - \mathbb{E}\left(f(X_n^i) \mid X_{n-1}^i\right)\right)$$

$$\Downarrow$$

Note $\mathbb{V}_n^N(f)$ is centered and when $\mathrm{osc}(f) \leq 1$ we have

$$\mathbb{E}\left(\left[\mathbb{V}_n^N(f)\right]^2 \mid (X_{n-1}^i)_i\right) = \int \eta_{n-1}^N(dx)\ P_n((f - P_n(f)(x))^2)(x) \leq 1$$

More compact perturbation formulas

$$\eta_n^{\mathbf{N}} = \eta_{n-1}^{\mathbf{N}} P_n + \frac{\mathbf{1}}{\sqrt{\mathbf{N}}}\ \mathbb{V}_n^N$$

equivalently

$$\eta_n^{\mathbf{N}} = \Phi_n\left(\eta_{n-1}^{\mathbf{N}}\right) + \frac{\mathbf{1}}{\sqrt{\mathbf{N}}}\ \mathbb{V}_n^N \quad \longrightarrow_{N\to\infty} \quad \eta_n = \Phi_n\left(\eta_{n-1}\right)$$

Fixed point linear integral equations (prescribed/targets (Boltzmann-Gibbs/Conditional laws)/...)

Homogenous models $(\Phi_n, P_n) = (\Phi, P) \rightsquigarrow$ **fixed point linear equation:**

$$\eta_n = \Phi(\eta_{n-1}) = \eta_{n-1}P = \eta_0 P^n \longrightarrow_{n\uparrow\infty} \quad \eta_\infty = \Phi(\eta_\infty) = \eta_\infty P$$

Fixed point linear integral equations (prescribed/targets (Boltzmann-Gibbs/Conditional laws)/...)

Homogenous models $(\Phi_n, P_n) = (\Phi, P) \rightsquigarrow$ fixed point linear equation:

$$\eta_n = \Phi(\eta_{n-1}) = \eta_{n-1}P = \eta_0 P^n \longrightarrow_{n\uparrow\infty} \quad \eta_\infty = \Phi(\eta_\infty) = \eta_\infty P$$

Solved with occupation measure/long run of the chain X_n:

$$\frac{1}{n}\sum_{0\leq k<n} f(X_k) \simeq_{n\uparrow\infty} \int f(x)\,\eta_\infty(dx)$$

Fixed point linear integral equations (prescribed/targets (Boltzmann-Gibbs/Conditional laws)/...)

Homogenous models $(\Phi_n, P_n) = (\Phi, P) \rightsquigarrow$ fixed point linear equation:

$$\eta_n = \Phi(\eta_{n-1}) = \eta_{n-1} P = \eta_0 P^n \longrightarrow_{n\uparrow\infty} \quad \eta_\infty = \Phi(\eta_\infty) = \eta_\infty P$$

Solved with occupation measure/long run of the chain X_n:

$$\frac{1}{n} \sum_{0 \leq k < n} f(X_k) \simeq_{n\uparrow\infty} \int f(x)\, \eta_\infty(dx)$$

Example $\eta_\infty = \mathcal{N}(0,1)$ Gaussian (reversible) shaker ($\oplus$ restricted to A)

$$X_n = \sqrt{1-\epsilon}\, X_{n-1} + \sqrt{\epsilon}\, W_n \qquad \text{with } \perp \text{ iid } W_n \sim \mathcal{N}(0,1)$$

Nonlinear integral equations

Solving nonlinear measure valued equations:

$$\eta_n = \Phi_n(\eta_{n-1}) = \eta_{n-1} P_{n,\eta_{n-1}} = \mathrm{Law}(\overline{X}_n)$$

with a (non unique) Markov chain interpretation

$$\overline{X}_n \sim P_{n,\eta_{n-1}}(\overline{X}_{n-1}, dx_n)$$

Nonlinear integral equations

Solving nonlinear measure valued equations:

$$\eta_n = \Phi_n\left(\eta_{n-1}\right) = \eta_{n-1} P_{n,\eta_{n-1}} = \mathrm{Law}(\overline{X}_n)$$

with a (non unique) Markov chain interpretation

$$\overline{X}_n \sim P_{n,\eta_{n-1}}(\overline{X}_{n-1}, dx_n)$$

Example

$$\overline{X}_n = \int F_n(x_{n-1}, \overline{X}_{n-1}, W_n)\ \mathbb{P}(\overline{X}_{n-1} \in dx_{n-1})$$

Example: How to sample $\overline{X}_1$ given $\overline{X}_0 \sim \eta_0(dx_0)$?

$$\overline{X}_1 \quad = \quad \int a(\overline{X}_0, x_0)\ \eta_0(dx_0) + \ W_1$$

Example: How to sample $\overline{X}_1$ given $\overline{X}_0 \sim \eta_0(dx_0)$?

$$\overline{X}_1 \quad = \quad \int a(\overline{X}_0, x_0)\; \eta_0(dx_0) + \; W_1$$

Example: How to sample $\overline{X}_1$ given $\overline{X}_0 \sim \eta_0(dx_0)$?

$$\begin{aligned}\overline{X}_1 &= \int a(\overline{X}_0, x_0)\ \eta_0(dx_0) + W_1 \\ &\overset{ex}{=} \int \sqrt{\log\left(1 + \|\overline{X}_0 - x_0\|^2\right)}\ \eta_0(dx_0) + W_1\end{aligned}$$

How to sample X_1 given $X_0 \sim \eta_0(dx_0)$?

$$X_1 = \int a(X_0, x_0)\ \eta_0(dx_0) + W_1$$

Sol. based on (ξ_0^i, W^i) iid copies of $(\overline{X}_0, W_1)$?

How to sample X_1 given $X_0 \sim \eta_0(dx_0)$?

$$X_1 = \int a(X_0, x_0)\ \eta_0(dx_0) + W_1$$

Sol. based on (ξ_0^i, W^i) iid copies of $(\overline{X}_0, W_1)$?

$$\xi_1^i = \int a(\xi_0^i, x_0)\ \eta_0^{\mathbf{N}}(dx_0) + W_1^i \quad \text{with} \quad \eta_0^{\mathbf{N}} := \frac{1}{N} \sum_{1 \leq j \leq N} \delta_{\xi_0^j}$$

How to sample X_1 given $X_0 \sim \eta_0(dx_0)$?

$$X_1 = \int a(X_0, x_0)\ \eta_0(dx_0) + \ W_1$$

Sol. based on (ξ_0^i, W^i) iid copies of $(\overline{X}_0, W_1)$?

$$\xi_1^i = \int a(\xi_0^i, x_0)\ \eta_0^{\mathbf{N}}(dx_0) + \ W_1^i \quad \text{with} \quad \eta_0^{\mathbf{N}} := \frac{1}{N} \sum_{1 \leq j \leq N} \delta_{\xi_0^j}$$

$$\Updownarrow$$

$$\xi_1^i = \frac{1}{N} \sum_{1 \leq j \leq N} a(\xi_0^i, \xi_0^{\mathbf{j}}) + \ W_1^i$$

[$\subset$ Mean field particle sampler]

How to sample X_1 given $X_0 \sim \eta_0(dx_0)$?

$$X_1 = \int a(X_0, x_0)\ \eta_0(dx_0) + W_1$$

Sol. based on (ξ_0^i, W^i) iid copies of $(\overline{X}_0, W_1)$?

$$\xi_1^i = \int a(\xi_0^i, x_0)\ \eta_0^{\mathbf{N}}(dx_0) + W_1^i \quad \text{with} \quad \eta_0^{\mathbf{N}} := \frac{1}{N} \sum_{1 \leq j \leq N} \delta_{\xi_0^j}$$

$$\Updownarrow$$

$$\xi_1^i = \frac{1}{N} \sum_{1 \leq j \leq N} a(\xi_0^i, \xi_0^{\mathbf{j}}) + W_1^i$$

[$\subset$ Mean field particle sampler]

Note: Running cost N^2 and ξ_1^i NOT iid

How to sample X_1 given $X_0 \sim \eta_0(dx_0)$?

$$X_1 = \int a(X_0, x_0)\ \eta_0(dx_0) + W_1$$

Sol. based on (ξ_0^i, W^i) iid copies of $(\overline{X}_0, W_1)$?

$$\xi_1^i = \int a(\xi_0^i, x_0)\ \eta_0^{\mathbf{N}}(dx_0) + W_1^i \quad \text{with} \quad \eta_0^{\mathbf{N}} := \frac{1}{N} \sum_{1 \leq j \leq N} \delta_{\xi_0^j}$$

$$\Updownarrow$$

$$\xi_1^i = \frac{1}{N} \sum_{1 \leq j \leq N} a(\xi_0^i, \xi_0^{\mathbf{j}}) + W_1^i$$

[$\subset$ Mean field particle sampler]

Note: Running cost N^2 and ξ_1^i NOT iid
but "almost iid" (propagation (initial) chaos)...

A brief review on sampling $\overline{X}_{t+dt}$ given $\overline{X}_t \sim \eta_t(dx_t)$

Continuous time version = McKean-Vlasov/Interacting diffusions

$$d\overline{X}_t = \overline{X}_{t+dt} - \overline{X}_t = \left(\int b(\overline{X}_t, x_t)\ \eta_t(dx_t) \right)\ dt + \underbrace{\sqrt{dt}\ N(0,1)}_{W_{t+dt} - W_t = dW_t}$$

A brief review on sampling $\overline{X}_{t+dt}$ given $\overline{X}_t \sim \eta_t(dx_t)$

Continuous time version = McKean-Vlasov/Interacting diffusions

$$d\overline{X}_t = \overline{X}_{t+dt} - \overline{X}_t = \left(\int b(\overline{X}_t, x_t)\ \eta_t(dx_t) \right)\ dt + \underbrace{\sqrt{dt}\ N(0,1)}_{W_{t+dt}-W_t=dW_t}$$

Continuous/Discrete time versions : Nonlinear/Interacting diffusions/jumps/accept-reject nonlinear Markov chains,. . .

A brief review on sampling $\overline{X}_{t+dt}$ given $\overline{X}_t \sim \eta_t(dx_t)$

Continuous time version = McKean-Vlasov/Interacting diffusions

$$d\overline{X}_t = \overline{X}_{t+dt} - \overline{X}_t = \left(\int b(\overline{X}_t, x_t)\ \eta_t(dx_t)\right)\ dt + \underbrace{\sqrt{dt}\ N(0,1)}_{W_{t+dt}-W_t=dW_t}$$

Continuous/Discrete time versions : Nonlinear/Interacting diffusions/jumps/accept-reject nonlinear Markov chains,. . .

$\rightsquigarrow$ **Particle filters, GA, SMC, DMC,. . . , EnKF,. . .**

N-interacting Markov chains

$$\xi^i_{n+1} \;\sim\; P_{n+1,\eta^{\mathbf{N}}_n}(\xi^i_n, dx_{n+1}) \quad \text{with} \quad \eta^{\mathbf{N}}_n = m(\xi_n) := \frac{1}{N}\sum_{1\leq i\leq N}\delta_{\xi^i_n} \;\simeq_{N\uparrow\infty}\; \eta_n$$

N-interacting Markov chains

$$\xi^i_{n+1} \sim P_{n+1,\eta^{\mathbf{N}}_n}(\xi^i_n, dx_{n+1}) \quad \text{with} \quad \eta^{\mathbf{N}}_n = m(\xi_n) := \frac{1}{N} \sum_{1 \leq i \leq N} \delta_{\xi^i_n} \simeq_{N\uparrow\infty} \eta_n$$

$$\Downarrow$$

Perturbation formulation

$$\eta^{\mathbf{N}}_n = \Phi_n\left(\eta^{\mathbf{N}}_{n-1}\right) + \frac{\mathbf{1}}{\sqrt{\mathbf{N}}}\, \mathbb{V}^{\mathbf{N}}_n \quad \longrightarrow_{\mathbf{N}\to\infty} \quad \eta_n = \Phi_n\left(\eta_{n-1}\right)$$

N-interacting Markov chains

$$\xi^i_{n+1} \sim P_{n+1,\eta^{\mathbf{N}}_n}(\xi^i_n, dx_{n+1}) \quad \text{with} \quad \eta^{\mathbf{N}}_n = m(\xi_n) := \frac{1}{N}\sum_{1\leq i\leq N} \delta_{\xi^i_n} \simeq_{N\uparrow\infty} \eta_n$$

$$\Downarrow$$

Perturbation formulation

$$\eta^{\mathbf{N}}_n = \Phi_n\left(\eta^{\mathbf{N}}_{n-1}\right) + \frac{\mathbf{1}}{\sqrt{\mathbf{N}}}\, \mathbb{V}^{\mathbf{N}}_n \quad \longrightarrow_{\mathbf{N}\to\infty} \quad \eta_n = \Phi_n\left(\eta_{n-1}\right)$$

$\forall f$ **/Multivariate/Functional/Donsker theorems**

$$(\mathbb{V}^{\mathbf{N}}_n)_{n\geq 0} \longrightarrow_{N\to\infty} (\mathbb{V}_n)_{n\geq 0}$$

with independent centered Gaussian fields $\mathbb{V}_n$ s.t.

$$\mathbb{E}\left([\mathbb{V}_n(f)]^2\right) = \int \eta_{n-1}(dx)\; P_{n,\eta_{n-1}}\left(\left(f - P_{n,\eta_{n-1}}(f)(x)\right)^2\right)(x)$$

Fixed point nonlinear equation (quasi-invariant,...)

Homogenous models $(\Phi, P_{n,\eta}) = (\Phi, P_{\eta}) \rightsquigarrow$ **fixed point nonlinear equation:**

$$\eta_n = \Phi(\eta_{n-1}) = \eta_{n-1} P_{\eta_{n-1}} \longrightarrow_{n\uparrow\infty} \quad \eta_\infty = \Phi(\eta_\infty) = \eta_\infty P_{\eta_\infty}$$

$$\Downarrow$$

Fixed point nonlinear equation (quasi-invariant,. . .)

Homogenous models $(\Phi, P_{n,\eta}) = (\Phi, P_\eta) \rightsquigarrow$ fixed point nonlinear equation:

$$\eta_n = \Phi(\eta_{n-1}) = \eta_{n-1} P_{\eta_{n-1}} \longrightarrow_{n\uparrow\infty} \quad \eta_\infty = \Phi(\eta_\infty) = \eta_\infty P_{\eta_\infty}$$

$$\Downarrow$$

Space/Time approximations:

$$\eta_n^{\mathbf{N}} \simeq_{N\uparrow\infty} \eta_n \simeq_{n\uparrow\infty} \eta_\infty \quad \text{or (when possible)} \quad \eta_n^{\mathbf{N}} \simeq_{n\uparrow\infty} \eta_\infty^N \simeq_{N\uparrow\infty} \eta_\infty$$

Fixed point nonlinear equation (quasi-invariant,. . .)

Homogenous models $(\Phi, P_{n,\eta}) = (\Phi, P_\eta)$ $\rightsquigarrow$ **fixed point nonlinear equation:**

$$\eta_n = \Phi(\eta_{n-1}) = \eta_{n-1} P_{\eta_{n-1}} \longrightarrow_{n\uparrow\infty} \quad \eta_\infty = \Phi(\eta_\infty) = \eta_\infty P_{\eta_\infty}$$

$$\Downarrow$$

Space/Time approximations:

$$\eta_n^{\mathbf{N}} \simeq_{N\uparrow\infty} \eta_n \simeq_{n\uparrow\infty} \eta_\infty \quad \text{or (when possible)} \quad \eta_n^{\mathbf{N}} \simeq_{n\uparrow\infty} \eta_\infty^N \simeq_{N\uparrow\infty} \eta_\infty$$

$\rightsquigarrow$ **Ergodic - space-time occupation estimates**

$$\frac{1}{n} \sum_{0\leq k<n} \eta_k^N \simeq_{(N,n)\uparrow\infty} \eta_\infty$$

Self interacting Markov chains - Fixed points nonlinear eq.

$$\eta_\infty = \Phi(\eta_\infty) = \eta_\infty P_{\eta_\infty}$$

Self interacting Markov chains - Fixed points nonlinear eq.

$$\eta_\infty = \Phi(\eta_\infty) = \eta_\infty P_{\eta_\infty}$$

Self interacting Markov chains

$$X_n \sim \ P_{\eta^n}(X_{n-1}, dx_n) \quad \text{with} \quad \eta^n := \frac{1}{n} \sum_{0 \leq k < n} \delta_{X_k} \simeq_{n \uparrow \infty} \eta_\infty$$

Self interacting Markov chains - Fixed points nonlinear eq.

$$\eta_\infty = \Phi(\eta_\infty) = \eta_\infty P_{\eta_\infty}$$

Self interacting Markov chains

$$X_n \sim P_{\eta^n}(X_{n-1}, dx_n) \quad \text{with} \quad \eta^n := \frac{1}{n} \sum_{0 \leq k < n} \delta_{X_k} \simeq_{n\uparrow\infty} \eta_\infty$$

Some references:

- Urn-models a.e. cv but no rates, Aldous-Flannery-Palacios CUP (1988).
- General Self-interacting + (sharp) rates (LSP-preprint 2002/SAA-2006), (Proc.Royal-Soc 2004) $\rightsquigarrow$ slides Oxford Univ. (2008).

- **Toy example in Self-interacting (LSP2002) & (Royal-Soc 2004)**

$\supset$ Elephant Random Walk (Schütz-Trimper-2004)

$$\Phi(\eta) = \epsilon\ \eta + (1-\epsilon)\ \underbrace{N(0,1)}_{=\eta_\infty} \quad (\Longrightarrow \Phi(\eta) - \Phi(\mu) = \epsilon\ (\eta - \mu))$$

Markov Chain Monte Carlo methods (MCMC)

$=$ **Design a Markov chain with prescribed invariant measure**

Markov Chain Monte Carlo methods (MCMC)

$=$ **Design a Markov chain with prescribed invariant measure**

Example: Find a Markov chain with prescribed reversible measure

$\Longleftrightarrow$ **Given π, find a Markov transition $P(x, dz)$ s.t.**

$$\pi(dx)P(x, dy) = \pi(dy)P(y, dx)$$

Metropolis-Hasting (last century top ten algo)

- Proposal transition/local search moves $K(x, dy) = \mathbb{P}(x \rightsquigarrow dy)$

Metropolis-Hasting (last century top ten algo)

- Proposal transition/local search moves $K(x, dy) = \mathbb{P}(x \rightsquigarrow dy)$
- Acceptance ratio

$$a(x, y) := \min\left(1, \frac{\pi(dy)K(y, dx)}{\pi(dx)K(x, dy)}\right)$$

Metropolis-Hasting (last century top ten algo)

- Proposal transition/local search moves $K(x, dy) = \mathbb{P}(x \rightsquigarrow dy)$
- Acceptance ratio

$$a(x,y) := \min\left(1, \frac{\pi(dy)K(y,dx)}{\pi(dx)K(x,dy)}\right)$$

$$\Updownarrow$$

$P(x, dz)$ = 2-step transition of the algorithm = $\mathbb{P}_{algo}(x \rightsquigarrow dz)$

$$x \overset{(1)}{\rightsquigarrow} y \sim K(x,dy) \overset{(2)}{\rightsquigarrow} z = \begin{cases} y & \text{with proba} \quad a(x,y) \\ x & \text{with proba} \quad 1 - a(x,y) \end{cases}$$

Metropolis-Hasting (last century top ten algo)

- Proposal transition/local search moves $K(x, dy) = \mathbb{P}(x \rightsquigarrow dy)$
- Acceptance ratio

$$a(x, y) := \min\left(1, \frac{\pi(dy)K(y, dx)}{\pi(dx)K(x, dy)}\right)$$

$$\Updownarrow$$

$P(x, dz)$ = 2-step transition of the algorithm = $\mathbb{P}_{algo}(x \rightsquigarrow dz)$

$$x \overset{(1)}{\rightsquigarrow} y \sim K(x, dy) \overset{(2)}{\rightsquigarrow} z = \begin{cases} y & \text{with proba} \quad a(x, y) \\ x & \text{with proba} \quad 1 - a(x, y) \end{cases}$$

Detailed balance/Master equation/Reversible property

$$\pi(dx)K(x, dy)\ a(x, y) = \min\left(\pi(dx)K(x, dy), \pi(dy)K(y, dx)\right) = \ldots$$

Gibbs sampler - ex. target on product spaces $x = (y, z)$

$$\begin{aligned} \pi(x) &= \pi(y, z) \\ &:= \mathbb{P}((Y, Z) = (y, z)) \end{aligned}$$

Gibbs sampler - ex. target on product spaces $x = (y, z)$

$$
\begin{aligned}
\pi(x) &= \pi(y,z) \\
&:= \mathbb{P}\left((Y,Z) = (y,z)\right) \\
&= \underbrace{\mathbb{P}\left(Y = y \mid Z = z\right)}_{=M_1(z,y)} \mathbb{P}\left(Z = z\right) = \underbrace{\mathbb{P}\left(Z = z \mid Y = y\right)}_{=M_2(y,z)} \mathbb{P}\left(Y = y\right)
\end{aligned}
$$

Gibbs sampler - ex. target on product spaces $x = (y, z)$

$$
\begin{aligned}
\pi(x) &= \pi(y,z) \\
&:= \mathbb{P}\left((Y,Z) = (y,z)\right) \\
&= \underbrace{\mathbb{P}\left(Y = y \mid Z = z\right)}_{=M_1(z,y)} \mathbb{P}\left(Z = z\right) = \underbrace{\mathbb{P}\left(Z = z \mid Y = y\right)}_{=M_2(y,z)} \mathbb{P}\left(Y = y\right)
\end{aligned}
$$

$$\Downarrow$$

Gibbs transition $P(x, x'') =$ **Two steps transition**

$$
x = \begin{pmatrix} y \\ z \end{pmatrix} \xrightarrow{y' \sim P_1(z,y')} x' := \begin{pmatrix} y' \\ z \end{pmatrix} \xrightarrow{z' \sim P_2(y',z')} x'' = \begin{pmatrix} y' \\ z' \end{pmatrix}
$$

Gibbs sampler - target on product spaces $x = (y, z)$

$\subset$ **Metropolis-Hasting !**

$$\text{Ex.:} \quad x \rightsquigarrow x' \;\sim\; P(x, x') = P((y, z), (y', z')) = P_1(z, y')\, 1_{z=z'}$$

Gibbs sampler - target on product spaces $x = (y, z)$

$\subset$ **Metropolis-Hasting !**

$$\text{Ex.:} \quad x \rightsquigarrow x' \;\sim\; P(x,x') = P((y,z),(y',z')) = P_1(z,y')\; 1_{z=z'}$$

$$\Downarrow$$

$$\frac{\pi(x')P(x',x)}{\pi(x)P(x,x')} = \frac{P_1(z',y')\mathbb{P}\left(Z=z'\right)1_{z'=z}P_1(z',y)}{P_1(z,y)\mathbb{P}\left(Z=z\right)1_{z=z'}P_1(z,y')} = 1$$

Exercises - Check shakers = MH chains

- **π_β-shakers - target density w.r.t. uniform-type ref. measure**

$$\pi_\beta(x) := \frac{1}{\mathcal{Z}_\beta}\, e^{-\beta\ U(x)} \quad \text{with sym. local prop.} \quad \mathbb{P}(x \rightsquigarrow y) = \mathbb{P}(y \rightsquigarrow x)$$

- **π_β-shakers with some reference measure**

$$\pi_\beta(dx) := \frac{1}{\mathcal{Z}_\beta}\, e^{-\beta\ U(x)} \ \times \underbrace{\nu(dx)}_{\text{example}=\mathcal{N}(0,1)}$$

with local propositions

$$\nu(dx) \times \mathbb{P}(x \rightsquigarrow dy) = \nu(dy) \times \mathbb{P}(y \rightsquigarrow dx)$$

- **π_A-shakers**

$$\pi_A(dx) := \frac{1}{\mathcal{Z}_A}\, 1_A(x)\ \underbrace{\nu(dx)}_{=\text{Law}(X)=\text{example}=\mathcal{N}(0,1)}$$

with local propositions

$$\nu(dx) \times \mathbb{P}(x \rightsquigarrow dy) = \nu(dy) \times \mathbb{P}(y \rightsquigarrow dx)$$

Boltzmann-Gibbs measures

Discrete time processes

(Linear) Markov chain Monte Carlo

Nonlinear Markov chain (Monte Carlo)

- A key Markov transport-formula
- Sequential Monte Carlo
- Feynman-Kac formulation

Updating/Boltzmann-Gibbs transf. $G(x) \geq 0$:

$$\eta \;:\; s.t.\;\; \eta(G) > 0 \quad \rightsquigarrow \quad \Psi_{\mathbf{G}}(\eta)(dx) := \frac{1}{\eta(G)}\; \mathbf{G}(\mathbf{x})\; \eta(\mathbf{dx})$$

Updating/Boltzmann-Gibbs transf. $G(x) \geq 0$:

$$\eta \; : \; s.t. \;\; \eta(G) > 0 \quad \rightsquigarrow \quad \Psi_{\mathbf{G}}(\eta)(dx) := \frac{1}{\eta(G)} \, \mathbf{G}(\mathbf{x}) \; \eta(\mathbf{dx})$$

Ex. of target Gibbs meas. with $U_n \geq U_{n-1}$ sequence of potential functions

$$\eta_n(dx) = \pi_{U_n}(dx) := \frac{1}{\nu(e^{-U_n})} \; e^{-U_n(x)} \; \nu(dx)$$

$$\Downarrow$$

$$\eta_n = \Psi_{G_{n-1}}(\eta_{n-1}) \quad \text{with} \quad G_{n-1} = e^{-(U_n - U_{n-1})} \in [0,1]$$

Updating/Boltzmann-Gibbs transf. $G(x) \geq 0$:

$$\eta \;:\; s.t.\;\; \eta(G) > 0 \quad \rightsquigarrow \quad \Psi_{\mathbf{G}}(\eta)(dx) := \frac{1}{\eta(G)}\; \mathbf{G}(\mathbf{x})\; \eta(\mathbf{dx})$$

Ex. of target Gibbs meas. with $U_n \geq U_{n-1}$ sequence of potential functions

$$\eta_n(dx) = \pi_{U_n}(dx) := \frac{1}{\nu(e^{-U_n})}\; e^{-U_n(x)}\; \nu(dx)$$

$$\Downarrow$$

$$\eta_n = \Psi_{G_{n-1}}(\eta_{n-1}) \quad \text{with} \quad G_{n-1} = e^{-(U_n - U_{n-1})} \in [0,1]$$

Equivalent formulation ($U_0 = 0$):

$$\eta_n(dx) \;\propto\; \left\{ \prod_{0 \leq p < n} G_p(x) \right\} \; \nu(dx)$$

Markov transport-formula

Updating/Boltzmann-Gibbs transformation

$$\Psi_G(\eta) = \eta S_{\eta,G}$$

with the (nonlinear+non unique)-Markov transition, for any $\epsilon_\eta G(x) \in [0,1]$

$$S_{\eta,G}(x,dy) = \epsilon_\eta G(x)\ \delta_x(dy) + (1 - \epsilon_\eta G(x))\ \Psi_G(\eta)(dy)$$

Examples:
$\epsilon_\eta = 0$ for any G;
$\epsilon_\eta = 1$ when $G \in [0,1]$;
$\epsilon_\eta = 1/\eta - essup - G \rightsquigarrow$ keep the best ...

Perfect sampler = Nonlinear Markov chain

Back to example $\eta_n = \pi_{U_n}$ **with some** U_n**-shaker** $\eta_n P_n = \eta_n$

Perfect sampler = Nonlinear Markov chain

Back to example $\eta_n = \pi_{U_n}$ **with some** U_n**-shaker** $\eta_n P_n = \eta_n$

$$\Downarrow$$

$$\eta_n = \eta_n P_n$$

Perfect sampler = Nonlinear Markov chain

Back to example $\eta_n = \pi_{U_n}$ **with some** U_n**-shaker** $\eta_n P_n = \eta_n$

$$\Downarrow$$

$$\eta_n = \eta_n P_n = \Psi_{G_{n-1}}(\eta_{n-1}) P_n$$

Perfect sampler = Nonlinear Markov chain

Back to example $\eta_n = \pi_{U_n}$ **with some** U_n**-shaker** $\eta_n P_n = \eta_n$

$$\Downarrow$$

$$\eta_n = \eta_n P_n = \Psi_{G_{n-1}}(\eta_{n-1}) P_n$$

$$\Downarrow$$

$$\eta_n = \eta_{n-1} P_{n,\eta_{n-1}} = \mathrm{Law}(\overline{X}_n)$$

with the transition of the **Perfect sampler** = Nonlinear Markov chain:

$$P_{n,\eta_{n-1}} = S_{\eta_{n-1},G_{n-1}} P_n$$

(Mean field) Interacting particle samplers (a.k.a. SMC)

Nonlinear Markov chain:

$$\eta_n = \eta_{n-1} P_{n,\eta_{n-1}} \quad \text{with} \quad P_{n,\eta_{n-1}} = S_{\eta_{n-1},G_{n-1}} P_n$$

$\rightsquigarrow$ **Mean field simulation = N-interacting Markov chains**

$$\xi_{n+1}^i \sim P_{n+1,\eta_n^{\mathbf{N}}}(\xi_n^i, dx_{n+1}) \Longrightarrow \eta_n^{\mathbf{N}} = m(\xi_n) := \frac{1}{N} \sum_{1 \leq i \leq N} \delta_{\xi_n^i} \simeq_{N\uparrow\infty} \eta_n$$

(Mean field) Interacting particle samplers (a.k.a. SMC)

Nonlinear Markov chain:

$$\eta_n = \eta_{n-1} P_{n,\eta_{n-1}} \quad \text{with} \quad P_{n,\eta_{n-1}} = S_{\eta_{n-1},G_{n-1}} P_n$$

$\rightsquigarrow$ **Mean field simulation $=$ N-interacting Markov chains**

$$\xi^i_{n+1} \sim P_{n+1,\eta^N_n}(\xi^i_n, dx_{n+1}) \Longrightarrow \eta^N_n = m(\xi_n) := \frac{1}{N} \sum_{1\leq i\leq N} \delta_{\xi^i_n} \simeq_{N\uparrow\infty} \eta_n$$

Note that

$$S_{\eta^N_n,G_n}(x,dy) = \epsilon_{\eta^N_n} G_n(x)\ \delta_x(dy) + (1-\epsilon_{\eta^N_n} G_n(x))\ \Psi_G(\eta^N_n)(dy)$$

and

$$\Psi_G(\eta^N_n) = \sum_{1\leq i\leq N} \frac{G_n(\xi^i_n)}{\sum_{1\leq j\leq N} G_n(\xi^j_n)}\ \delta_{\xi^i_n}$$

$\epsilon_{\eta^N_n} = 0 \rightsquigarrow$ simple genetic algorithm,. . . .

(Mean field) Interacting particle samplers (a.k.a. SMC)

Nonlinear Markov chain:

$$\eta_n = \eta_{n-1} P_{n,\eta_{n-1}} \quad \text{with} \quad P_{n,\eta_{n-1}} = S_{\eta_{n-1},G_{n-1}} P_n$$

$\rightsquigarrow$ **Mean field simulation $=$ N-interacting Markov chains**

$$\xi^i_{n+1} \sim P_{n+1,\eta^N_n}(\xi^i_n, dx_{n+1}) \Longrightarrow \eta^N_n = m(\xi_n) := \frac{1}{N} \sum_{1\leq i\leq N} \delta_{\xi^i_n} \simeq_{N\uparrow\infty} \eta_n$$

Note that

$$S_{\eta^N_n,G_n}(x,dy) = \epsilon_{\eta^N_n} G_n(x)\, \delta_x(dy) + (1-\epsilon_{\eta^N_n} G_n(x))\, \Psi_G(\eta^N_n)(dy)$$

and

$$\Psi_G(\eta^N_n) = \sum_{1\leq i\leq N} \frac{G_n(\xi^i_n)}{\sum_{1\leq j\leq N} G_n(\xi^j_n)}\, \delta_{\xi^i_n}$$

$\epsilon_{\eta^N_n} = 0 \rightsquigarrow$ simple genetic algorithm,. . . .

Inversely: Any GA cv to the above equation.

Feynman-Kac formulation - also works **for any** (G_n, P_n)

Nonlinear updating/Markov transport:

$$\eta_n = \Phi_n(\eta_{n-1})$$

Feynman-Kac formulation - also works **for any** (G_n, P_n)

Nonlinear updating/Markov transport:

$$\eta_n = \Phi_n(\eta_{n-1}) = \Psi_{G_{n-1}}(\eta_{n-1})P_n$$

$$\Updownarrow \quad \text{with} \quad \mathbf{P_n(x_{n-1}, dx_n)} := \mathbb{P}(\mathbf{X_n} \in \mathbf{dx_n} \mid \mathbf{X_{n-1}} = \mathbf{x_{n-1}})$$

FK solution/semigroup/formulation:

$$\eta_n(f) = \gamma_n(f)/\gamma_n(1) \quad \text{with} \quad \gamma_n(f) = \mathbb{E}\left(f(X_n) \prod_{0 \leq p < n} G_p(X_p)\right)$$

Feynman-Kac formulation - also works **for any** (G_n, P_n)

Nonlinear updating/Markov transport:

$$\eta_n = \Phi_n(\eta_{n-1}) = \Psi_{G_{n-1}}(\eta_{n-1}) P_n$$

$$\Updownarrow \quad \text{with} \quad \mathbf{P_n(x_{n-1}, dx_n)} := \mathbb{P}(\mathbf{X_n} \in \mathbf{dx_n} \mid \mathbf{X_{n-1}} = \mathbf{x_{n-1}})$$

FK solution/semigroup/formulation:

$$\eta_n(f) = \gamma_n(f)/\gamma_n(1) \quad \text{with} \quad \gamma_n(f) = \mathbb{E}\left(f(X_n) \prod_{0 \leq p < n} G_p(X_p) \right)$$

$\leftrightsquigarrow$ **Genetic algo./Monte Carlo samplers (mutation,selection)** $\sim (P_n, G_n)$

Feynman-Kac formulation - also works **for any** (G_n, P_n)

Nonlinear updating/Markov transport:

$$\eta_n = \Phi_n(\eta_{n-1}) = \Psi_{G_{n-1}}(\eta_{n-1})P_n$$

$$\Updownarrow \quad \text{with} \quad \mathbf{P_n(x_{n-1}, dx_n)} := \mathbb{P}(\mathbf{X_n} \in \mathbf{dx_n} \mid \mathbf{X_{n-1}} = \mathbf{x_{n-1}})$$

FK solution/semigroup/formulation:

$$\eta_n(f) = \gamma_n(f)/\gamma_n(1) \quad \text{with} \quad \gamma_n(f) = \mathbb{E}\left(f(X_n) \prod_{0 \le p < n} G_p(X_p)\right)$$

$\leadsto$ **Genetic algo./Monte Carlo samplers (mutation,selection)** $\sim (P_n, G_n)$

Linear/Gaussian models = Kalman filter 1960, (Swerling 1958), Finite state = Wonham filter (1964), ...