Evolution with Drifting Targets

Varun Kanade

(with Leslie Valiant and Jenn Wortman Vaughan)

Harvard University

Outline of Talk

Computational model for evolution

Drift and monotone evolution

Evolving hyperplanes and conjunctions

Drift-resistant and quasi-monotone evolvability

Evolution: Mutation & Natural Selection







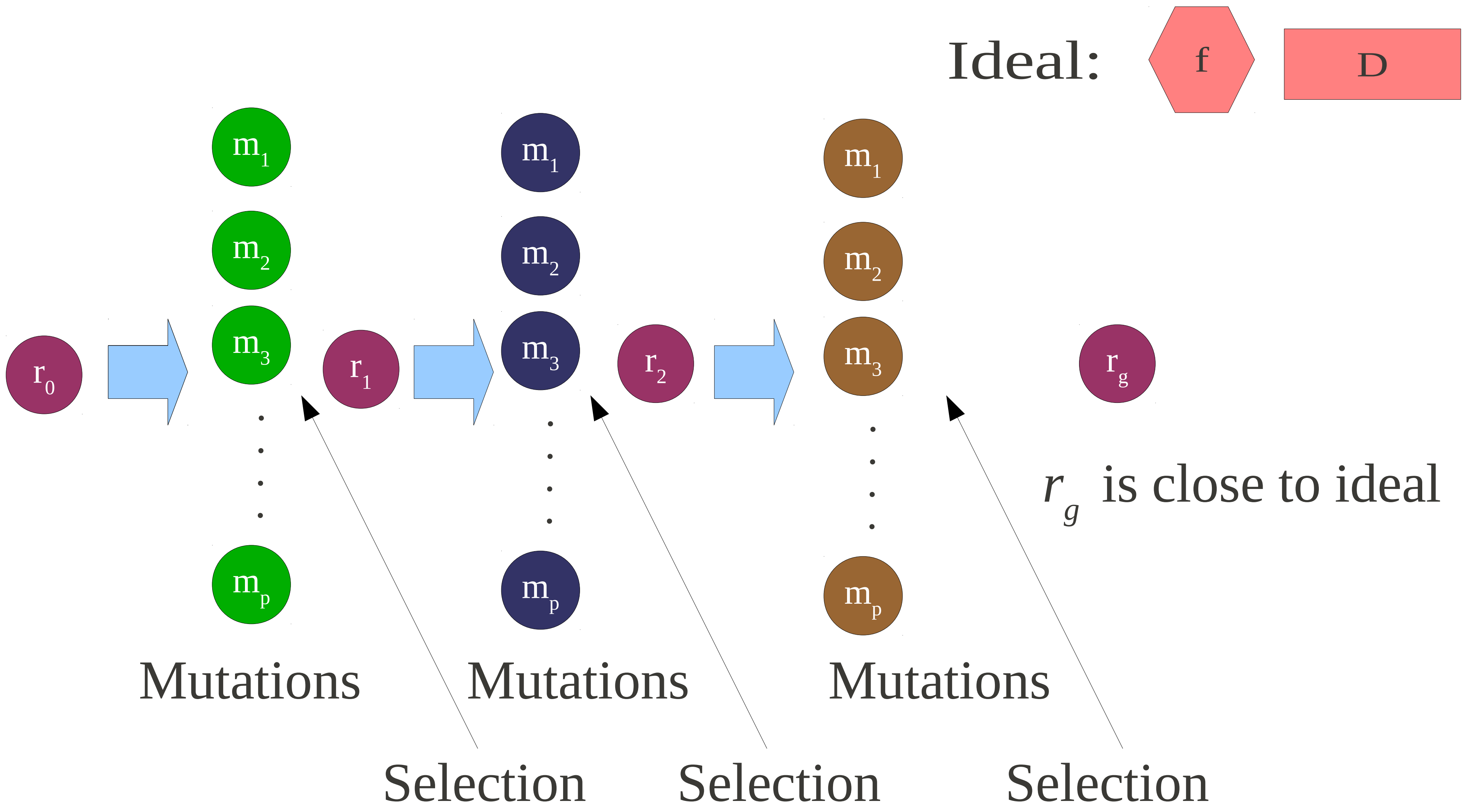

Computational Model

Evolve to ideal function for best behavior

Mutations at every generation

The fit members survive to the next generation

Computational Model



(Valiant 2007)

Modeling Mutation

Mutator: Poly-time probabilistic Turing Machine
Takes current representation r

$$r \rightarrow \{ (m_1, q_1), \ldots , (m_p, q_p) \}$$

Generates (polynomially many) mutations and probabilities of occurrence.

Performance: Ideal function f; target distribution D.

$$\mathrm{Perf}_D(r, f) = \mathbf{E}_D[\, r(x)\, f(x)]$$

Beneficial & Neutral Mutations

Evolutionary algorithm gets only empirical estimates of true performances
(S - poly-size sample of examples from D)

Mutation $r \rightarrow m$ is *beneficial* if

$$\mathrm{Perf}_S(m, f) \geq \mathrm{Perf}_S(r, f) + \tau$$

Mutation $r \rightarrow m$ is *neutral* if

$$|\mathrm{Perf}_S(m, f) - \mathrm{Perf}_S(r, f)| \leq \tau$$

Selection Rules

If there exists a *beneficial* mutation one is selected at random according to probability of occurrence

Otherwise, a *neutral* mutation is selected according to probability of occurrence

Concept class ***C* is evolvable under *D*** if for every target function $f \in C$, and every $\varepsilon > 0$ an evolutionary algorithm in $g(\varepsilon)$ generations reaches a representation r that has performance ($\mathbf{E}_D[r(x)f(x)]$) at least $1 - \varepsilon$, w.p. $\geq 1 - \varepsilon$.

Previous Work

Evolvable concepts subclass of SQ learnable concepts (Valiant 2007)

Evolvability of monotone conjunctions under uniform distribution (Valiant 2007)

Evolvability equivalent to CSQ learning (queries only ask for correlation with target) (Feldman 2008)

Robustness of Model: Several alternative definitions lead to the same model (Feldman 2009)

Drifting Targets

Organisms adapt to gradual changes in environment

Evolvability model should be robust to drift in ideal function

Evolutionary algorithm adapts to change in perpetuity

Modeling Drifting Targets

Distribution D

Target functions f_1 , f_2, f_3, ...

Small drift rate $\mathbf{E}_D[\,|f_i(x) - f_{i+1}(x)|\,] \leq \Delta$

Evolvable with Drift Δ

Start at r_0

There exists time g (polynomial) *s.t.* for every $i \geq g$, with probability at least $1 - \varepsilon$, $\mathrm{Perf}_D(r_i , f_i) \geq 1 - \varepsilon$

Main Result

All evolvable concept classes are also evolvable with drifting target ideal functions

Monotonic Evolution

Representations r_1 , r_2 , … of an evolutionary algorithm

Monotonic Evolution

Monotonic if for all i, with probability at least $1 - \varepsilon$

$$\mathrm{Perf}_D(r_i, f) \geq \mathrm{Perf}_D(r_{i-1}, f)$$

Strictly Monotonic Evolution (μ)

Strictly monotonic if for all i, with probability at least $1 - \varepsilon$

$$\mathrm{Perf}_D(r_i , f) \geq \mathrm{Perf}_D(r_{i-1}, f) + \mu$$

Beneficial Neighborhood

Neighbourhood: Set of mutations of *r*

Beneficial Neighborhood (μ): Neighbourhood containing at least one representation *r′* satisfying

$$\mathbf{Perf}_D(r', f) \geq \mathbf{Perf}_D(r, f) + \mu$$

Theorem: For a given concept class *C*, if there exists a set of representations such that there always exists a beneficial neighborhood (μ), then *C* is evolvable for drifting targets as long as drift $\Delta \leq \mu - 1/\text{poly}$

Evolving Halfspaces and Conjunctions

Evolving Halfspaces

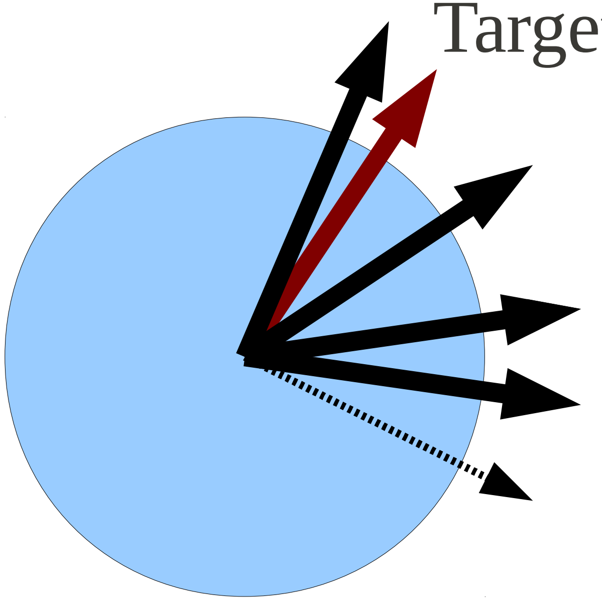
Algorithm for evolving halfspaces passing through the origin

For arbitrary distributions this is impossible (Feldman 2008)

Algorithm under symmetric distributions

Extend to product normal distributions

Evolving Hyperplanes



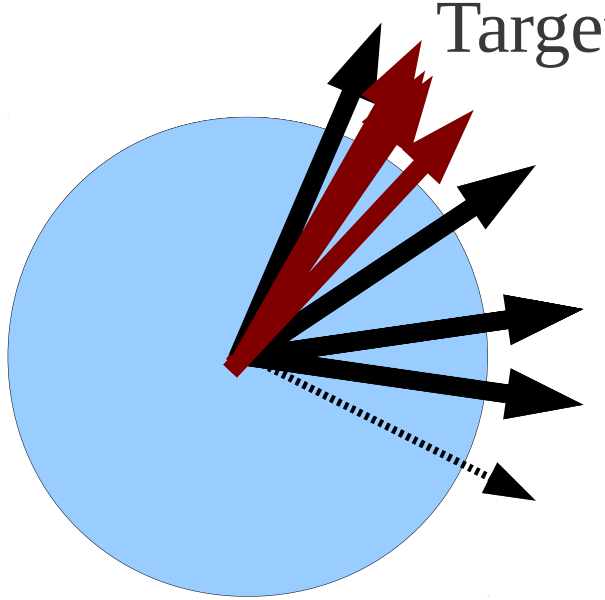
Mutations:

$\boldsymbol{r} \rightarrow \cos(\theta)\, \boldsymbol{r} + \sin(\theta)\, \boldsymbol{e}$

e is a unit vector of an orthogonal basis of which ***r*** is a part.

Tolerates drift of O(ε/n)

Evolving Hyperplanes



Mutations:

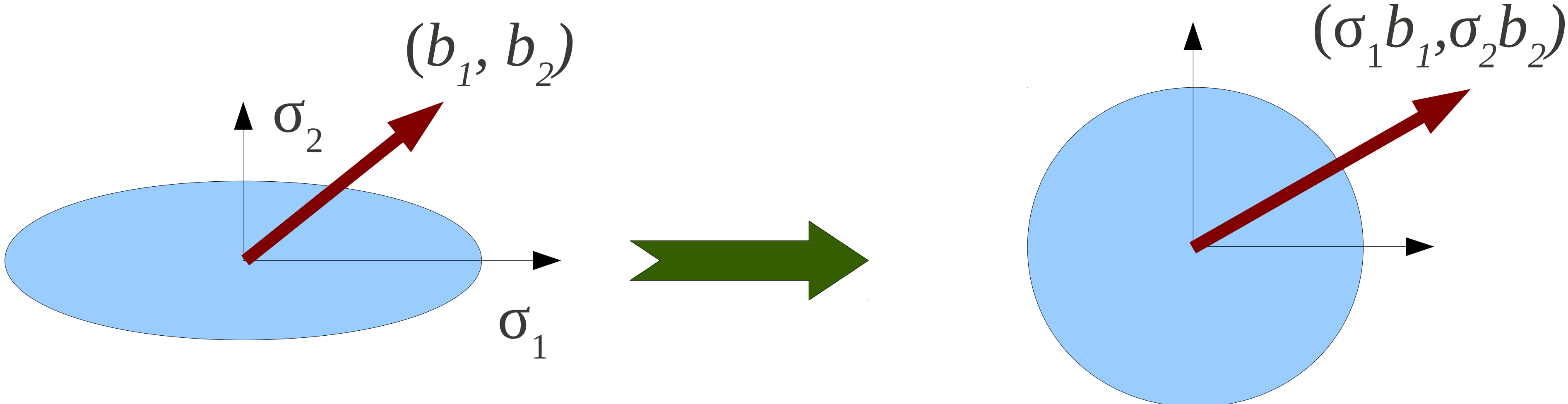
$\boldsymbol{r} \rightarrow \cos(\theta)\, \boldsymbol{r} + \sin(\theta)\, \boldsymbol{e}$

e is a unit vector of an orthogonal basis of which ***r*** is a part.

Tolerates drift of O(ε/n)

A Different Algorithm

Generalize to product normal distributions



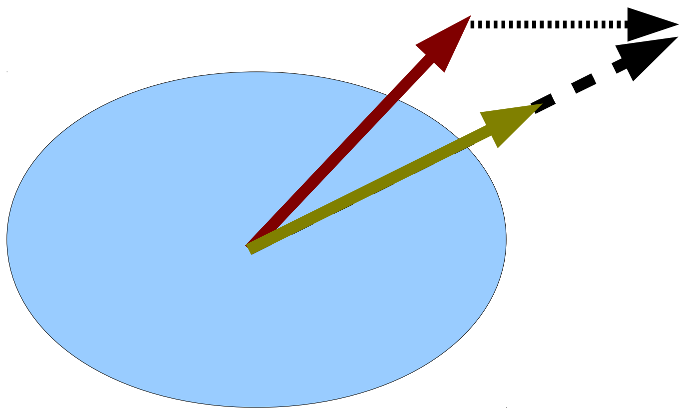
$(x_1, x_2) \rightarrow (x_1/\sigma_1, x_2/\sigma_2)$

Problem: We do not know σ_1 and σ_2. Evolutionary algorithm never sees actual examples, only sees the performance

Evolving Halfspaces

A different algorithm – adds a small component to each direction

Somewhat similar to rotation



Evolving Conjunctions

Monotonic conjunctions under uniform distribution over $\{0, 1\}^n$ (Valiant 2007)

Example: $x_1 \wedge x_7 \wedge x_{13}$

Mutations: Add a literal; drop a literal; swap a literal

Beneficial Neighborhood: $\mu = O(\varepsilon^2)$

Can generalize to all conjunctions (Jacobson 07)

Drift Resistance for Evolvability

Evolution with Drifting Targets

Can all evolutionary algorithms be made resistant to some drift?

Yes!

How much drift?

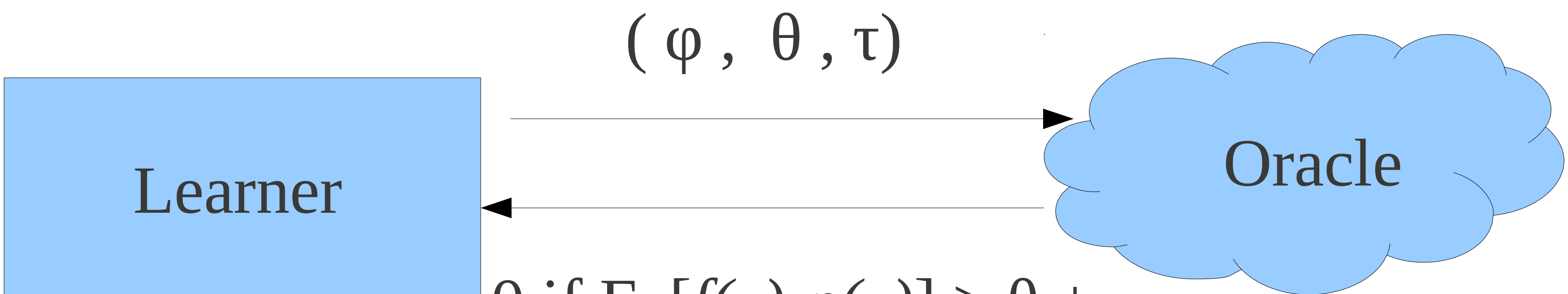
Small, but inverse polynomial

Can all evolutionary algorithms be made monotonic?

No, but can make quasi-monotonic

$\text{CSQ}_{>}$ Learning

Target function: f Distribution: D



0 if $E_D[f(x)\ \varphi(x)] \geq \theta + \tau$

1 if $E_D[\ f(x)\ \varphi(x)] \leq \theta - \tau$

Any of 0 or 1 otherwise

This is equivalent to correlational SQ (CSQ) learning (binary search)

(Feldman 2008)

Overview of Simulation

Feldman's simulation of $CSQ_{>}$ algorithm that makes q queries of tolerance τ

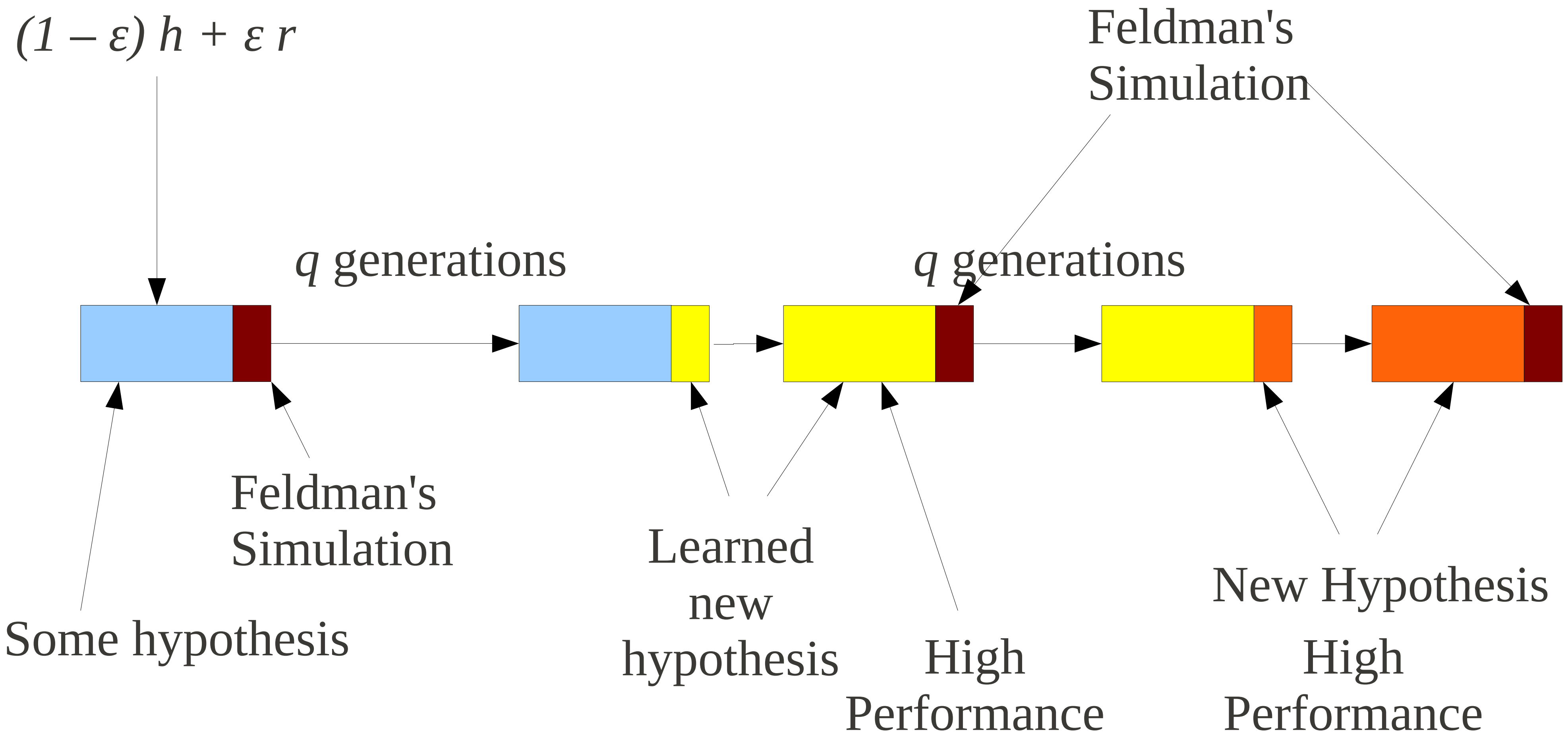
Hypothesis h output by $CSQ_{>}$ algorithm has high performance

Make drift small enough so that for q rounds of evolution answers don't change (up to tolerance)

But need evolutionary algorithm to run in perpetuity

(Feldman 2008)

Sketch of Reduction



Technical Problem: Need representation independent of ε – this requires a special construction

Evolution with Drifting Targets

All evolvable concept classes are also evolvable with drifting targets

All evolvable concept classes can be evolved quasi-monotonically

Give some drift rates for halfspaces through origin and conjunctions