A General Coarse-Graining Framework for Studying Simultaneous Inter-Population Constraints Induced by Evolutionary Operations

Keki Burjorjee & Jordan B. Pollack

DEMO Lab
Computer Science Department
Brandeis University

July 10, 2006

Inter-Population Constraints

- Assume there is some population in some evolutionary system (GA, GP, ES, etc.)
  - Call it p
- Suppose that either selection or variation is applied to p
  - Call the resulting population q

$$p \xrightarrow{\mathcal{W}} q$$

- What are the constraints on the composition of q?

Inter-Population Constraints

- Assume there is some population in some evolutionary system (GA, GP, ES, etc.)
  - Call it p
- Suppose that either selection or variation is applied to p
  - Call the resulting population q

$$p \xrightarrow{\mathcal{W}} q$$

- What are the constraints on the composition of q?

Inter-Population Constraints

$$p \xrightarrow{\mathcal{W}} q$$

- What are the constraints on the composition of q?
- Can q be *any* population?
- (Clearly) No. Its composition will be constrained by
  - The composition of p
  - Which evolutionary operation $\mathcal{W}$ is
- Terminology
  - p : pre-operative population of $\mathcal{W}$
  - q : post-operative population of $\mathcal{W}$
- An inter-population constraint is a constraint that an evolutionary operation induces between *any* pre and post operative populations

Inter-Population Constraints

$$p \xrightarrow{\mathcal{W}} q$$

- What are the constraints on the composition of q?
- Can q be *any* population?
- (Clearly) No. Its composition will be constrained by
  - The composition of p
  - Which evolutionary operation $\mathcal{W}$ is
- Terminology
  - p : pre-operative population of $\mathcal{W}$
  - q : post-operative population of $\mathcal{W}$
- An inter-population constraint is a constraint that an evolutionary operation induces between *any* pre and post operative populations

Inter-Population Constraints

$$p \xrightarrow{\mathcal{W}} q$$

- What are the constraints on the composition of q?
- Can q be *any* population?
- (Clearly) No. Its composition will be constrained by
  - The composition of p
  - Which evolutionary operation $\mathcal{W}$ is
- Terminology
  - p : pre-operative population of $\mathcal{W}$
  - q : post-operative population of $\mathcal{W}$
- An inter-population constraint is a constraint that an evolutionary operation induces between *any* pre and post operative populations

Inter-Population Constraints

$$p \xrightarrow{\mathcal{W}} q$$

- What are the constraints on the composition of q?
- Can q be *any* population?
- (Clearly) No. Its composition will be constrained by
  - The composition of p
  - Which evolutionary operation $\mathcal{W}$ is
- Terminology
  - p : pre-operative population of $\mathcal{W}$
  - q : post-operative population of $\mathcal{W}$
- An inter-population constraint is a constraint that an evolutionary operation induces between *any* pre and post operative populations

Inter-Population Constraints

$$p \xrightarrow{\mathcal{W}} q$$

- What are the constraints on the composition of q?
- Can q be *any* population?
- (Clearly) No. Its composition will be constrained by
  - The composition of p
  - Which evolutionary operation $\mathcal{W}$ is
- Terminology
  - p : pre-operative population of $\mathcal{W}$
  - q : post-operative population of $\mathcal{W}$
- An inter-population constraint is a constraint that an evolutionary operation induces between *any* pre and post operative populations

Schema Theoretic Approach to Deriving Inter-population Constraints

- The Schema Theoretic approach is to:
  1. Define a grammar for specifying schemata, then
  2. Derive a schema theorem that describes how evolutionary operations alter the frequencies of schemata
- The conditions for applicability of each schema theorem are very strict
- Each theory applicable *only* when
  - Genotypes are of a particular datastructure
  - Particular variation operators are used
  - Schemata are defined in a particular way
- The results of each schema theorem do *not* carry over when any of these change

Schema Theoretic Approach to Deriving Inter-population Constraints

- The Schema Theoretic approach is to:
  1. Define a grammar for specifying schemata, then
  2. Derive a schema theorem that describes how evolutionary operations alter the frequencies of schemata
- The conditions for applicability of each schema theorem are very strict
- Each theory applicable *only* when
  - Genotypes are of a particular datastructure
  - Particular variation operators are used
  - Schemata are defined in a particular way
- The results of each schema theorem do *not* carry over when any of these change

Schema Theoretic Approach to Deriving Inter-population Constraints

- The Schema Theoretic approach is to:
  1. Define a grammar for specifying schemata, then
  2. Derive a schema theorem that describes how evolutionary operations alter the frequencies of schemata
- The conditions for applicability of each schema theorem are very strict
- Each theory applicable *only* when
  - Genotypes are of a particular datastructure
  - Particular variation operators are used
  - Schemata are defined in a particular way
- The results of each schema theorem do *not* carry over when any of these change

Schema Theoretic Approach to Deriving Inter-population Constraints

- The Schema Theoretic approach is to:
  1. Define a grammar for specifying schemata, then
  2. Derive a schema theorem that describes how evolutionary operations alter the frequencies of schemata
- The conditions for applicability of each schema theorem are very strict
- Each theory applicable *only* when
  - Genotypes are of a particular datastructure
  - Particular variation operators are used
  - Schemata are defined in a particular way
- The results of each schema theorem do *not* carry over when any of these change

Schema Theoretic Approach to Deriving Inter-population Constraints

- The Schema Theoretic approach is to:
  1. Define a grammar for specifying schemata, then
  2. Derive a schema theorem that describes how evolutionary operations alter the frequencies of schemata
- The conditions for applicability of each schema theorem are very strict
- Each theory applicable *only* when
  - Genotypes are of a particular datastructure
  - Particular variation operators are used
  - Schemata are defined in a particular way
- The results of each schema theorem do *not* carry over when any of these change

Schema Theoretic Approach to Deriving Inter-population Constraints

- The Schema Theoretic approach is to:
    1. Define a grammar for specifying schemata, then
    2. Derive a schema theorem that describes how evolutionary operations alter the frequencies of schemata
- The conditions for applicability of each schema theorem are very strict
- Each theory applicable *only* when
    - Genotypes are of a particular datastructure
    - Particular variation operators are used
    - Schemata are defined in a particular way
- The results of each schema theorem do *not* carry over when any of these change

Schema Theoretic Approach to Deriving Inter-population Constraints

- The Schema Theoretic approach is to:
  1. Define a grammar for specifying schemata, then
  2. Derive a schema theorem that describes how evolutionary operations alter the frequencies of schemata
- The conditions for applicability of each schema theorem are very strict
- Each theory applicable *only* when
  - Genotypes are of a particular datastructure
  - Particular variation operators are used
  - Schemata are defined in a particular way
- The results of each schema theorem do *not* carry over when any of these change

A Different Technique for Deriving Inter-Population Constraints

- Form Invariant Commutation: a different technique for deriving inter-population constraints
- Uses coarse-grainings
- A coarse-graining is just a function from the genotype set to some set
  - e.g. if G is a genotype set and K is some set then $\beta : G \longrightarrow K$ is a coarse-graining

A Different Technique for Deriving Inter-Population Constraints

- Form Invariant Commutation: a different technique for deriving inter-population constraints
- Uses coarse-grainings
- A coarse-graining is just a function from the genotype set to some set
  - e.g. if G is a genotype set and K is some set then $\beta : G \longrightarrow K$ is a coarse-graining

A Different Technique for Deriving Inter-Population Constraints

- Form Invariant Commutation: a different technique for deriving inter-population constraints
- Uses coarse-grainings
- A coarse-graining is just a function from the genotype set to some set
  - e.g. if G is a genotype set and K is some set then $\beta : G \longrightarrow K$ is a coarse-graining

A Different Technique for Deriving Inter-Population Constraints

- Form Invariant Commutation: a different technique for deriving inter-population constraints
- Uses coarse-grainings
- A coarse-graining is just a function from the genotype set to some set
    - e.g. if G is a genotype set and K is some set then $\beta : G \rightarrow K$ is a coarse-graining

An Abstract Coarse-Graining Framework

- Present a framework that uses form invariant commutation to obtain theorems about the inter-population constraints induced by evolutionary operations
- Framework is *abstract* in that unlike various schema theories applicability of theorems in this framework is *not* limited by
  - a specific genotypic datastructure, or
  - specific variation operators, or
  - a specific grammar for defining schemata
- Rather applicability of results is limited by whether an abstract relationship exists between the variation operators and coarse-grainings used
  - Relationship is called Ambivalence

An Abstract Coarse-Graining Framework

- Present a framework that uses form invariant commutation to obtain theorems about the inter-population constraints induced by evolutionary operations
- Framework is *abstract* in that unlike various schema theories applicability of theorems in this framework is *not* limited by
  - a specific genotypic datastructure, or
  - specific variation operators, or
  - a specific grammar for defining schemata
- Rather applicability of results is limited by whether an abstract relationship exists between the variation operators and coarse-grainings used
  - Relationship is called Ambivalence

An Abstract Coarse-Graining Framework

- Present a framework that uses form invariant commutation to obtain theorems about the inter-population constraints induced by evolutionary operations
- Framework is *abstract* in that unlike various schema theories applicability of theorems in this framework is *not* limited by
  - a specific genotypic datastructure, or
  - specific variation operators, or
  - a specific grammar for defining schemata
- Rather applicability of results is limited by whether an abstract relationship exists between the variation operators and coarse-grainings used
  - Relationship is called Ambivalence

An Abstract Coarse-Graining Framework

- Present a framework that uses form invariant commutation to obtain theorems about the inter-population constraints induced by evolutionary operations
- Framework is *abstract* in that unlike various schema theories applicability of theorems in this framework is *not* limited by
  - a specific genotypic datastructure, or
  - specific variation operators, or
  - a specific grammar for defining schemata
- Rather applicability of results is limited by whether an abstract relationship exists between the variation operators and coarse-grainings used
  - Relationship is called Ambivalence

An Abstract Coarse-Graining Framework

- Present a framework that uses form invariant commutation to obtain theorems about the inter-population constraints induced by evolutionary operations
- Framework is *abstract* in that unlike various schema theories applicability of theorems in this framework is *not* limited by
  - a specific genotypic datastructure, or
  - specific variation operators, or
  - a specific grammar for defining schemata
- Rather applicability of results is limited by whether an abstract relationship exists between the variation operators and coarse-grainings used
  - Relationship is called Ambivalence

An Abstract Coarse-Graining Framework

- Present a framework that uses form invariant commutation to obtain theorems about the inter-population constraints induced by evolutionary operations
- Framework is *abstract* in that unlike various schema theories applicability of theorems in this framework is *not* limited by
  - a specific genotypic datastructure, or
  - specific variation operators, or
  - a specific grammar for defining schemata
- Rather applicability of results is limited by whether an abstract relationship exists between the variation operators and coarse-grainings used
  - Relationship is called Ambivalence

An Abstract Coarse-Graining Framework

- Present a framework that uses form invariant commutation to obtain theorems about the inter-population constraints induced by evolutionary operations
- Framework is *abstract* in that unlike various schema theories applicability of theorems in this framework is *not* limited by
  - a specific genotypic datastructure, or
  - specific variation operators, or
  - a specific grammar for defining schemata
- Rather applicability of results is limited by whether an abstract relationship exists between the variation operators and coarse-grainings used
  - Relationship is called Ambivalence

Benefits of this Framework

- Given some
  - fitness proportional evolutionary system, and some
  - coarse-graining

  Easy to determine whether this framework is useful for deriving inter-population constraints
  - Just check if the ambivalence relationship holds
- Determination of whether schema analysis is useful typically proceeds by "trying to do the schema analysis"
- Once it is determined that this framework is useful, burden of analysis is greatly reduced
  - The "grunt work" is done by the "machinary" of the framework
- Framework accommodates multi-parent variation operators in a natural way
- For the set of fixed length bitstring GAs with 'common' variation operations
  - General framework reduces to a specific framework for analyzing the effect of one evolutionary step on the marginal distributions of populations

Benefits of this Framework

- Given some
  - fitness proportional evolutionary system, and some
  - coarse-graining

  Easy to determine whether this framework is useful for deriving inter-population constraints
  - Just check if the ambivalence relationship holds
- Determination of whether schema analysis is useful typically proceeds by "trying to do the schema analysis"
- Once it is determined that this framework is useful, burden of analysis is greatly reduced
  - The "grunt work" is done by the "machinary" of the framework
- Framework accommodates multi-parent variation operators in a natural way
- For the set of fixed length bitstring GAs with 'common' variation operations
  - General framework reduces to a specific framework for analyzing the effect of one evolutionary step on the marginal distributions of populations

Benefits of this Framework

- Given some
  - fitness proportional evolutionary system, and some
  - coarse-graining

  Easy to determine whether this framework is useful for deriving inter-population constraints
  - Just check if the ambivalence relationship holds
- Determination of whether schema analysis is useful typically proceeds by "trying to do the schema analysis"
- Once it is determined that this framework is useful, burden of analysis is greatly reduced
  - The "grunt work" is done by the "machinary" of the framework
- Framework accommodates multi-parent variation operators in a natural way
- For the set of fixed length bitstring GAs with 'common' variation operations
  - General framework reduces to a specific framework for analyzing the effect of one evolutionary step on the marginal distributions of populations

Benefits of this Framework

- Given some
  - fitness proportional evolutionary system, and some
  - coarse-graining

  Easy to determine whether this framework is useful for deriving inter-population constraints
  - Just check if the ambivalence relationship holds
- Determination of whether schema analysis is useful typically proceeds by "trying to do the schema analysis"
- Once it is determined that this framework is useful, burden of analysis is greatly reduced
  - The "grunt work" is done by the "machinary" of the framework
- Framework accommodates multi-parent variation operators in a natural way
- For the set of fixed length bitstring GAs with 'common' variation operations
  - General framework reduces to a specific framework for analyzing the effect of one evolutionary step on the marginal distributions of popululations

Benefits of this Framework

- Given some
  - fitness proportional evolutionary system, and some
  - coarse-graining

  Easy to determine whether this framework is useful for deriving inter-population constraints
  - Just check if the ambivalence relationship holds
- Determination of whether schema analysis is useful typically proceeds by "trying to do the schema analysis"
- Once it is determined that this framework is useful, burden of analysis is greatly reduced
  - The "grunt work" is done by the "machinary" of the framework
- Framework accommodates multi-parent variation operators in a natural way
- For the set of fixed length bitstring GAs with 'common' variation operations
  - General framework reduces to a specific framework for analyzing the effect of one evolutionary step on the marginal distributions of populations

Modeling Genotypic Variation With Transmission Functions

- Use a transmission functions (Altenberg 1994) to model any stochastic variation function
  - that takes n genotypes as parents, and
  - produces 1 genotype as a child
- Example: $T(g|g_1, \ldots g_n)$ is the probability that some variation operation produces g as a child given parents $g_1, \ldots, g_n$,

Modeling Genotypic Variation With Transmission Functions

- Use a transmission functions (Altenberg 1994) to model any stochastic variation function
  - that takes n genotypes as parents, and
  - produces 1 genotype as a child
- Example: $T(g|g_1, \ldots g_n)$ is the probability that some variation operation produces g as a child given parents $g_1, \ldots, g_n$,

Modeling Populations and Operations on Populations

- Populations modeled as real valued distributions over the genotype set.
  - Distribution values sum to 1
- Evolutionary operations are modeled as *parameterized* mathematical operators
  - Take genotypic distributions as input and produce genotypic distributions as output
  - Parameter objects used by the operator in the calculation of output

Modeling the Effect of Variation on Populations

- Effect of any variation function modeled by T on some population p is given by the variation operator $\mathcal{V}_T$

If $p' = V_T(p)$, then p' is as follows: For any genotype g,

$$p'(g) = \sum_{\substack{(g_1,\ldots,g_m) \\ \in \prod_1^m G}} T(g|g_1,\ldots,g_m) \prod_{i=1}^{m} p(g_i)$$

Modeling the Effect of Selection on Populations

- Effect of fitness proportional selection on some population p using any fitness function f given by the selection operator $\mathcal{S}_f$

If $p' = S_f(p)$, then p' is as follows: For any genotype g,

$$p'(g) = \frac{f(g)p(g)}{\mathcal{E}_f(p)}$$

where $\mathcal{E}_f$ is the weighted average fitness of p

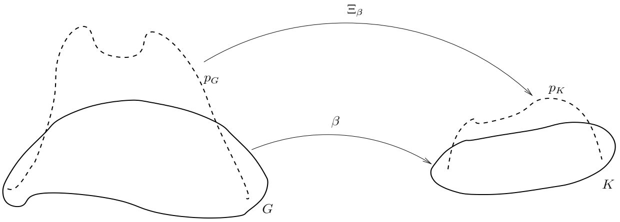
Coarse-graining Terminology and Notation

- For any coarsegraining $\beta : G \rightarrow K$
  - Call co-domain K the β-theme set
  - Call the elements of K β-themes

- For any $g \in G$, $k \in K$ such that $\beta(g) = k$, say that g β-instantiates k
  - $\langle k \rangle_\beta$ denotes the set of all $g \in G$ that β-instantiate k
  - Call $\langle k \rangle_\beta$ the β-theme class of k

Projection Operator

- Let $\beta : G \longrightarrow K$ be a coarsegraining
- A projection operator Ξ_β 'projects' a distribution p_G over G 'through' β to create a distribution $p_K = \Xi_\beta(p_G)$ over the theme set

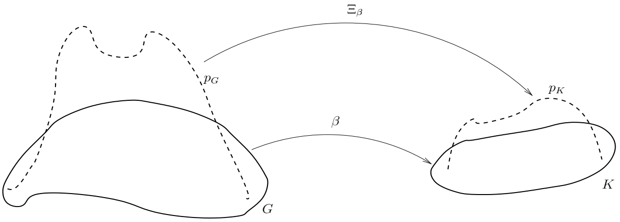


For any $k \in K$,

$$p_K(k) = \sum_{g \in \langle k \rangle_\beta} p(g)$$

Projection Operator

- Let $\beta : G \rightarrow K$ be a coarsegraining
- A projection operator Ξ_β 'projects' a distribution p_G over G 'through' β to create a distribution $p_K = \Xi_\beta(p_G)$ over the theme set



For any $k \in K$,

$$p_K(k) = \sum_{g \in \langle k \rangle_\beta} p(g)$$

A Technique for Obtaining Inter-Population Constraints

Form Invariant Commutation

- $\beta : G \rightarrow K$ some coarsegraining
- $\mathcal{W}$ a parameterizable operator parameterized by some object x such that
- for any population p_G such that
- Then we've obtained a single inter population constraint for the operator $\mathcal{W}_x$
- Call y the *quotient parameter*
  Okay for the y quotient parameter to depend on p_G
  - As long as the nature of this dependence is well understood

A single inter-population constraint is good . . .

A Technique for Obtaining Inter-Population Constraints

Form Invariant Commutation

- $\beta : G \rightarrow K$ some coarsegraining
- $\mathcal{W}$ a parameterizable operator parameterized by some object x such that
- for any population p_G such that

- Then we've obtained a single inter population constraint for the operator $\mathcal{W}_x$
- Call y the *quotient parameter*
  Okay for the y quotient parameter to depend on p_G
  - As long as the nature of this dependence is well understood

A single inter-population constraint is good . . .

A Technique for Obtaining Inter-Population Constraints

Form Invariant Commutation

- $\beta : G \to K$ some coarsegraining
- $\mathcal{W}$ a parameterizable operator parameterized by some object x such that
- for any population p_G such that

$$p_G \xrightarrow{\mathcal{W}_x} p'_G$$

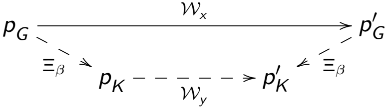
- Then we've obtained a single inter population constraint for the operator $\mathcal{W}_x$
- Call y the *quotient parameter*
  Okay for the y quotient parameter to depend on p_G
  - As long as the nature of this dependence is well understood

A single inter-population constraint is good . . .

A Technique for Obtaining Inter-Population Constraints

Form Invariant Commutation

- $\beta : G \to K$ some coarsegraining
- $\mathcal{W}$ a parameterizable operator parameterized by some object x such that
- for any population p_G such that



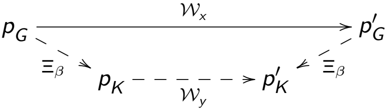
- Then we've obtained a single inter population constraint for the operator $\mathcal{W}_x$
- Call y the *quotient parameter*
  Okay for the y quotient parameter to depend on p_G
  - As long as the nature of this dependence is well understood

A single inter-population constraint is good . . .

A Technique for Obtaining Inter-Population Constraints

Form Invariant Commutation

- $\beta : G \to K$ some coarsegraining
- $\mathcal{W}$ a parameterizable operator parameterized by some object x such that
- for any population p_G such that



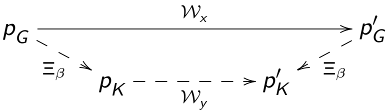
- Then we've obtained a single inter population constraint for the operator $\mathcal{W}_x$
- Call y the *quotient parameter*
  Okay for the y quotient parameter to depend on p_G
  - As long as the nature of this dependence is well understood

A single inter-population constraint is good ...

A Technique for Obtaining Inter-Population Constraints

Form Invariant Commutation

- $\beta : G \rightarrow K$ some coarsegraining
- $\mathcal{W}$ a parameterizable operator parameterized by some object x such that
- for any population p_G such that



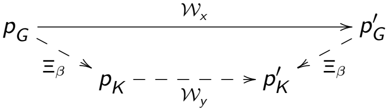
- Then we've obtained a single inter population constraint for the operator $\mathcal{W}_x$
- Call y the *quotient parameter*
  Okay for the y quotient parameter to depend on p_G
  - As long as the nature of this dependence is well understood

A single inter-population constraint is good . . .

A Technique for Obtaining Inter-Population Constraints

Form Invariant Commutation

- $\beta : G \rightarrow K$ some coarsegraining
- $\mathcal{W}$ a parameterizable operator parameterized by some object x such that
- for any population p_G such that



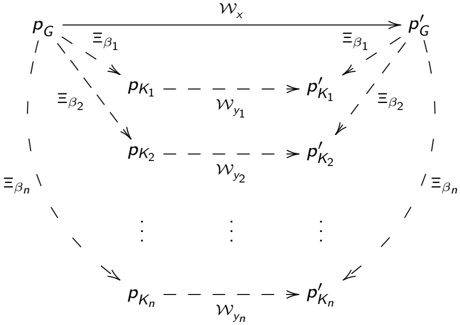
- Then we've obtained a single inter population constraint for the operator $\mathcal{W}_x$
- Call y the *quotient parameter*
  Okay for the y quotient parameter to depend on p_G
  - As long as the nature of this dependence is well understood

A single inter-population constraint is good . . .

A Technique for Obtaining Inter-Population Constraints

Form Invariant Commutation

- More inter-population constraints give more information.

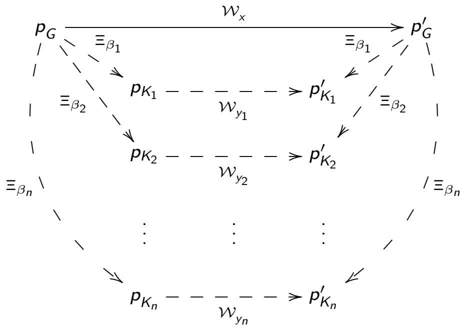


- if $\beta_1, \ldots, \beta_n$ induce different partitions over the genotype set
  - Call these *simultaneous* constraints

A Technique for Obtaining Inter-Population Constraints

Form Invariant Commutation

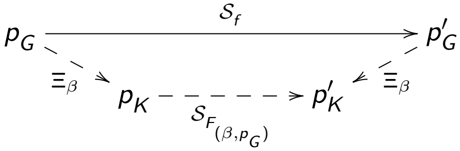
- More inter-population constraints give more information.



- if $\beta_1, \ldots, \beta_n$ induce different partitions over the genotype set
  - Call these *simultaneous* constraints

Selectional Constraints Theorem

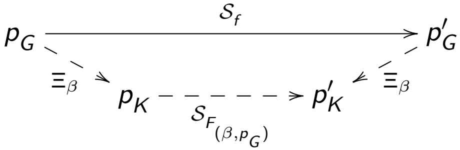
For *any* coarse-graining $\beta : G \rightarrow K$ and any population p_G,



- $F_{(\beta,p_G)} : K \rightarrow \mathbb{R}^+$ is called the β-*theme fitness function* of p_G
- It assigns to each theme $k \in K$ the weighted average fitness of all genotypes that β-instantiate k
- $F_{(\beta,p_G)}$ depends on p_G, but that's okay since we understand the (simple) nature of this dependence

Selectional Constraints Theorem

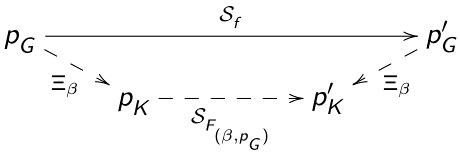
For *any* coarse-graining $\beta : G \to K$ and any population p_G,



- $F_{(\beta,p_G)} : K \to \mathbb{R}^+$ is called the *β-theme fitness function* of p_G
- It assigns to each theme $k \in K$ the weighted average fitness of all genotypes that β-instantiate k
- $F_{(\beta,p_G)}$ depends on p_G, but that's okay since we understand the (simple) nature of this dependence

Selectional Constraints Theorem

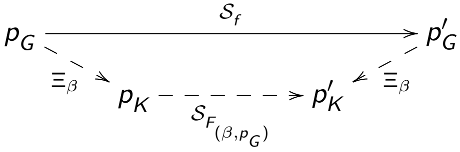
For *any* coarse-graining $\beta : G \rightarrow K$ and any population p_G,



- $F_{(\beta,p_G)} : K \rightarrow \mathbb{R}^+$ is called the β-*theme fitness function* of p_G
- It assigns to each theme $k \in K$ the weighted average fitness of all genotypes that β-instantiate k
- $F_{(\beta,p_G)}$ depends on p_G, but that's okay since we understand the (simple) nature of this dependence

Selectional Constraints Theorem

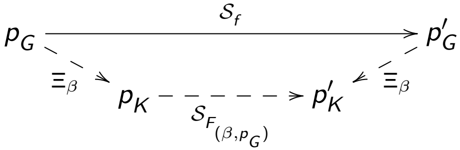
For *any* coarse-graining $\beta : G \rightarrow K$ and any population p_G,



- $F_{(\beta,p_G)} : K \rightarrow \mathbb{R}^+$ is called the β-*theme fitness function* of p_G
- It assigns to each theme $k \in K$ the weighted average fitness of all genotypes that β-instantiate k
- $F_{(\beta,p_G)}$ depends on p_G, but that's okay since we understand the (simple) nature of this dependence

Selectional Constraints Theorem

For *any* coarse-graining $\beta : G \rightarrow K$ and any population p_G,



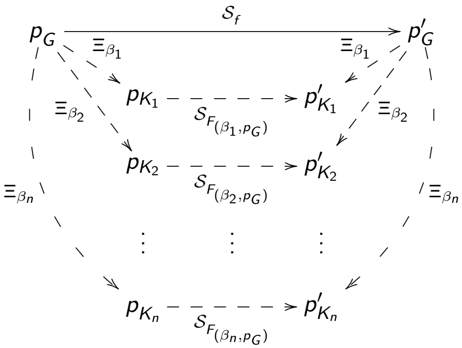
- $F_{(\beta,p_G)} : K \rightarrow \mathbb{R}^+$ is called the β-*theme fitness function* of p_G
- It assigns to each theme $k \in K$ the weighted average fitness of all genotypes that β-instantiate k
- $F_{(\beta,p_G)}$ depends on p_G, but that's okay since we understand the (simple) nature of this dependence

Simultaneous Selectional Constraints

- Result of the selection constraints theorem is valid for *any* coarse-graining
- So its easy to obtain simultaneous selectional constraints
- For any coarsegrainings $\beta_1, \ldots, \beta_n$, and any population p_G,

Simultaneous Selectional Constraints

- Result of the selection constraints theorem is valid for *any* coarse-graining
- So its easy to obtain simultaneous selectional constraints
- For any coarsegrainings $\beta_1, \ldots, \beta_n$, and any population p_G,



What about Variation?

- Inter-population constraints easy to derive for the selection operator
  - *Any* coarse-graining can be used
- Matters not so simple in the case of variation
- Introduce a relationship between a transmission function and a coarse-graining called *ambivalence*
- When ambivalence is satisfied derivation of an inter-population variational constraint using the coarse-graining is possible

What about Variation?

- Inter-population constraints easy to derive for the selection operator
  - *Any* coarse-graining can be used
- Matters not so simple in the case of variation
- Introduce a relationship between a transmission function and a coarse-graining called *ambivalence*
- When ambivalence is satisfied derivation of an inter-population variational constraint using the coarse-graining is possible

What about Variation?

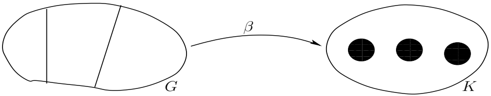
- Inter-population constraints easy to derive for the selection operator
  - *Any* coarse-graining can be used
- Matters not so simple in the case of variation
- Introduce a relationship between a transmission function and a coarse-graining called *ambivalence*
- When ambivalence is satisfied derivation of an inter-population variational constraint using the coarse-graining is possible

What about Variation?

- Inter-population constraints easy to derive for the selection operator
  - *Any* coarse-graining can be used
- Matters not so simple in the case of variation
- Introduce a relationship between a transmission function and a coarse-graining called *ambivalence*
- When ambivalence is satisfied derivation of an inter-population variational constraint using the coarse-graining is possible

Ambivalence by Example

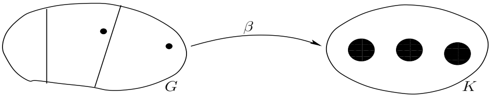
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

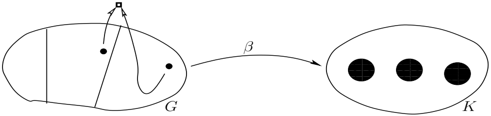
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

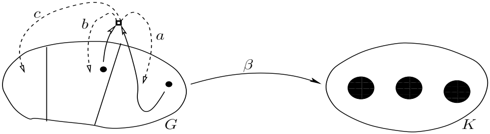
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

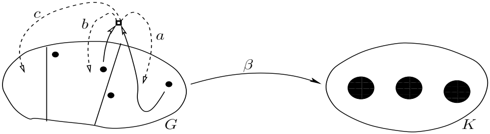
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

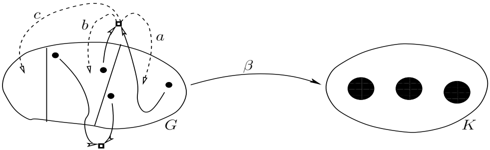
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

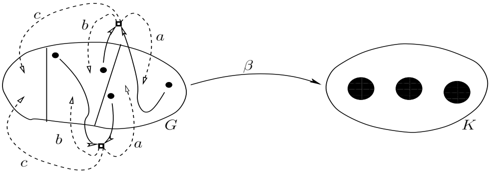
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

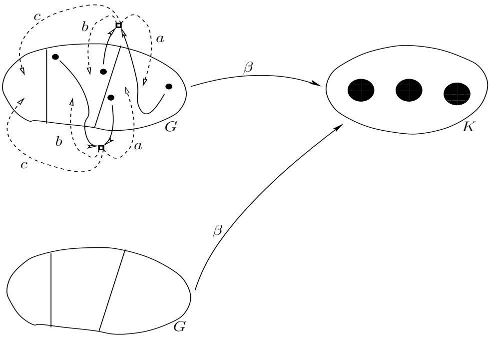
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

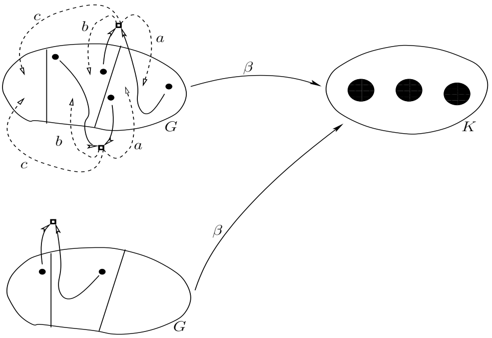
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

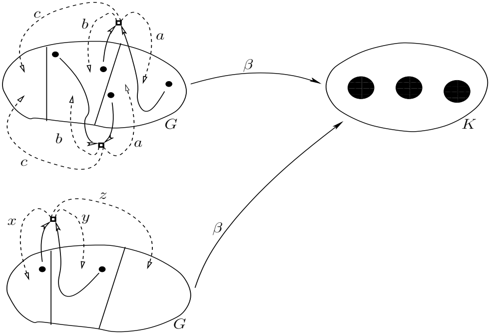
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

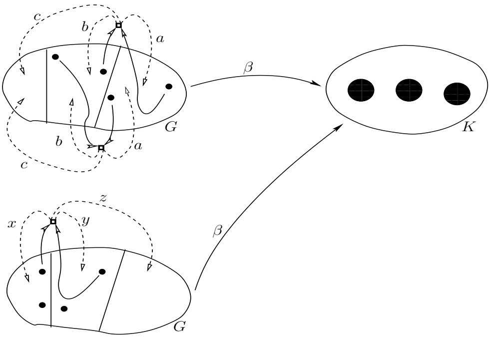
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

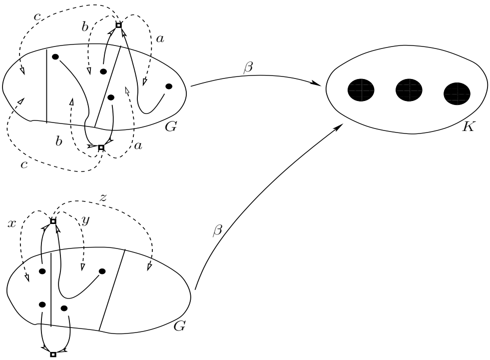
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

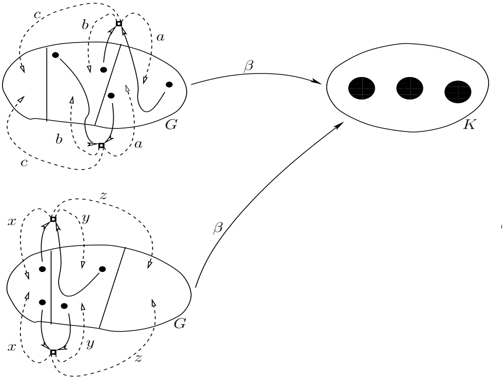
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

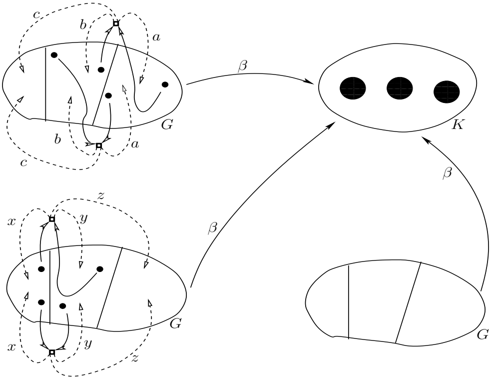
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

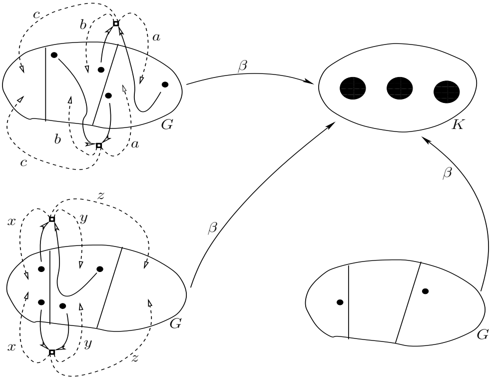
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

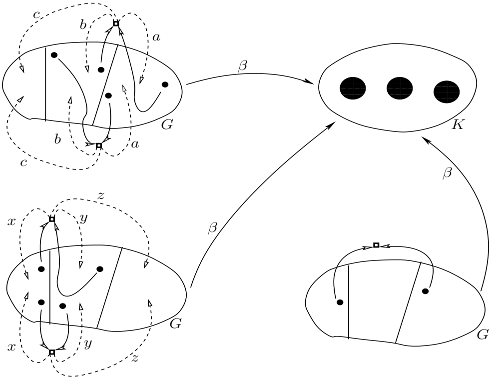
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

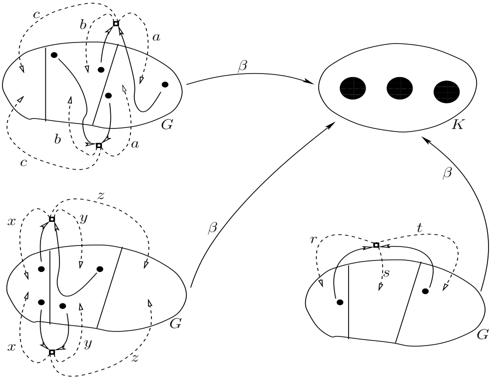
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

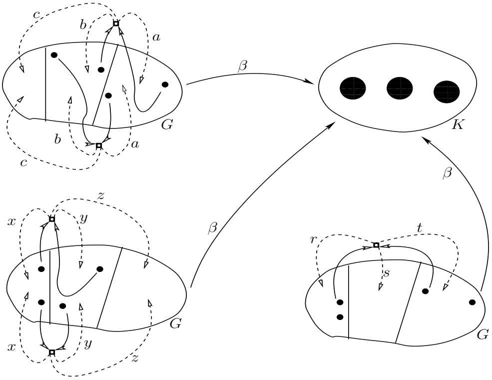
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

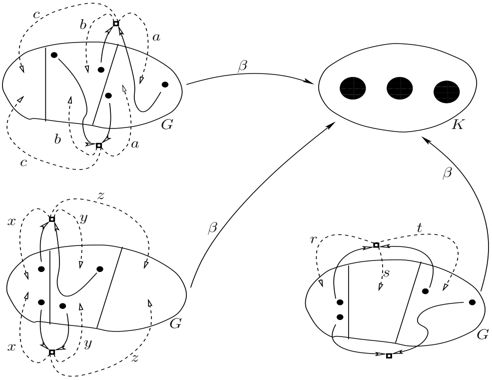
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

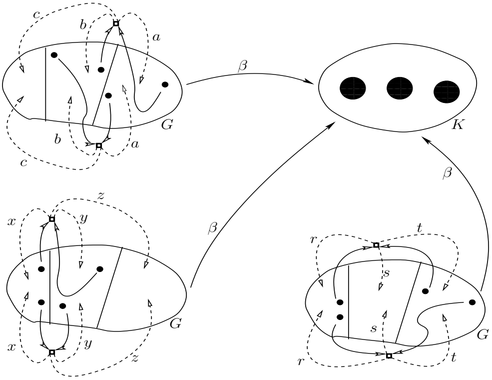
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

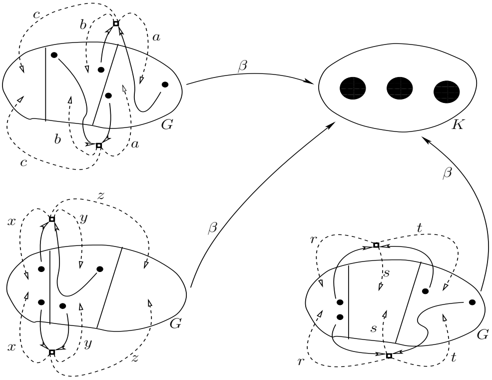
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

Ambivalence by Example

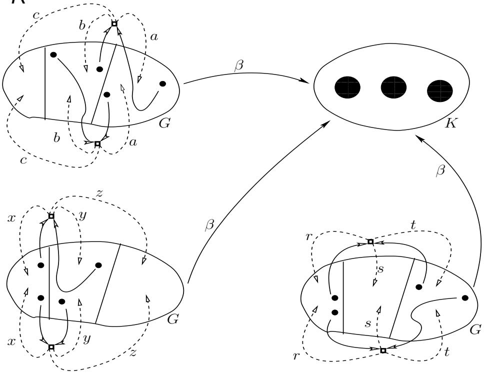
An Ambivalent 2-parent transmission function T



Say that T is *ambivalent under* β

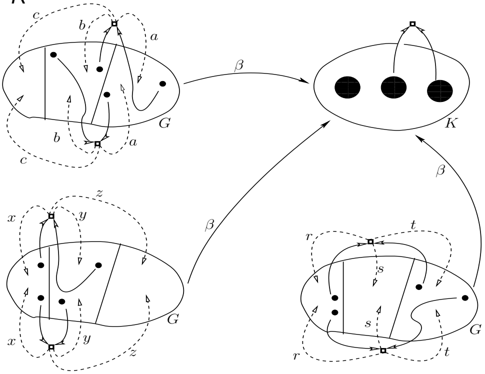
Projection of an Ambivalent Transmission function

- T an ambivalent under some coarse-graining $\beta : G \rightarrow K$,
- β-projection of T, denoted $T^{\overrightarrow{\beta}}$, is a transmission function over K



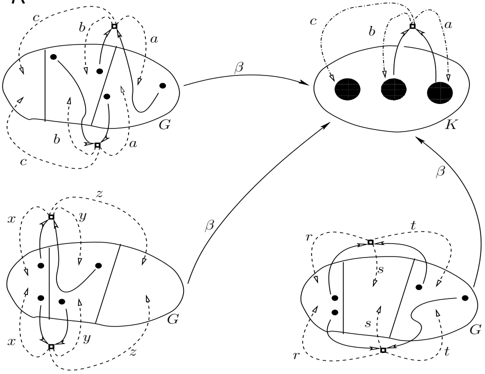
Projection of an Ambivalent Transmission function

- T an ambivalent under some coarse-graining $\beta : G \to K$,
- β-projection of T, denoted $T^{\overrightarrow{\beta}}$, is a transmission function over K



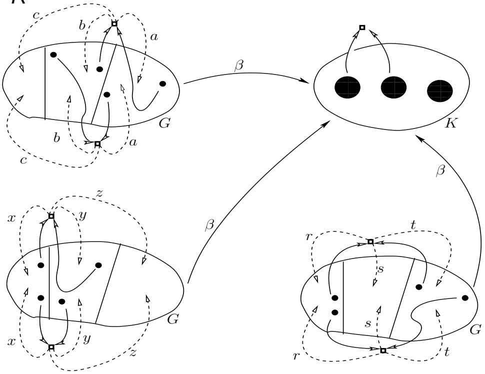
Projection of an Ambivalent Transmission function

- T an ambivalent under some coarse-graining $\beta : G \rightarrow K$,
- β-projection of T, denoted $T^{\overrightarrow{\beta}}$, is a transmission function over K



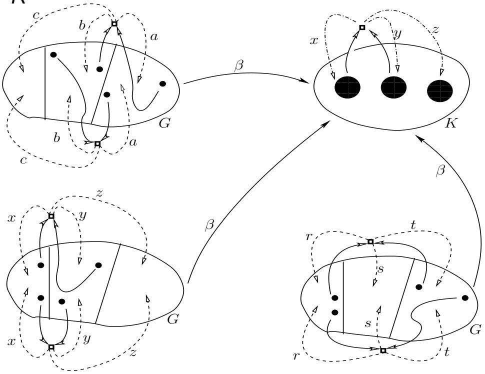
Projection of an Ambivalent Transmission function

- T an ambivalent under some coarse-graining $\beta : G \to K$,
- β-projection of T, denoted $T^{\overrightarrow{\beta}}$, is a transmission function over K



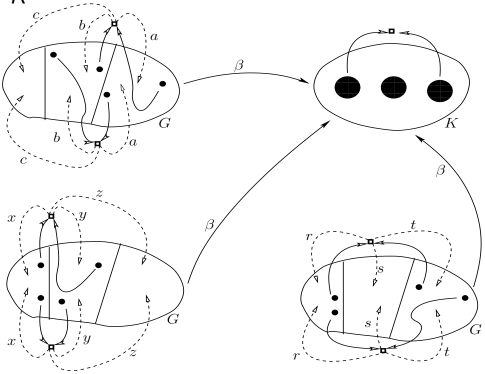
Projection of an Ambivalent Transmission function

- T an ambivalent under some coarse-graining $\beta : G \to K$,
- β-projection of T, denoted $T^{\overrightarrow{\beta}}$, is a transmission function over K



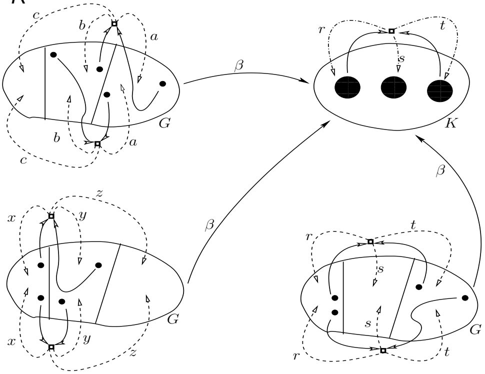
Projection of an Ambivalent Transmission function

- T an ambivalent under some coarse-graining $\beta : G \rightarrow K$,
- β-projection of T, denoted $T^{\overrightarrow{\beta}}$, is a transmission function over K

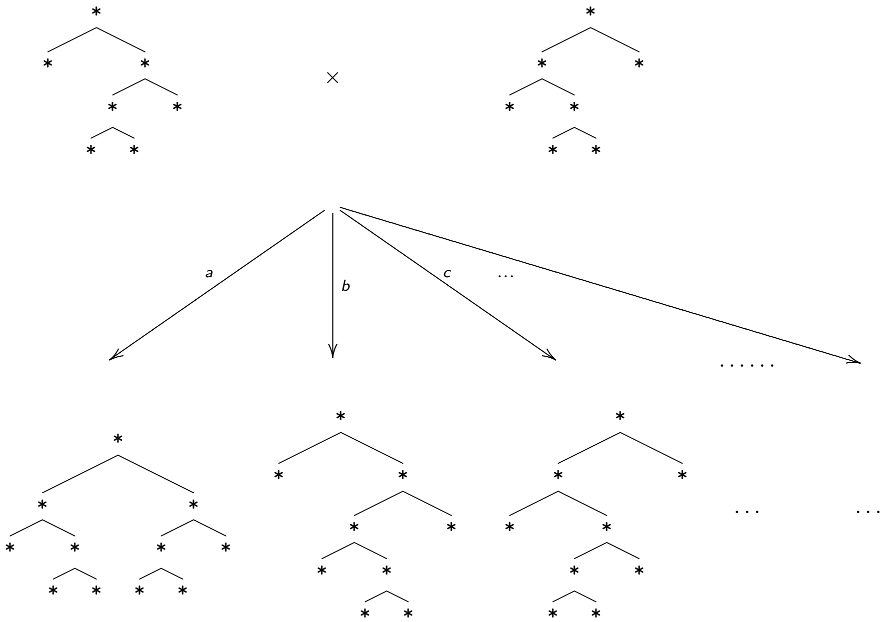


Projection of an Ambivalent Transmission function

- T an ambivalent under some coarse-graining $\beta : G \to K$,
- β-projection of T, denoted $T^{\overrightarrow{\beta}}$, is a transmission function over K

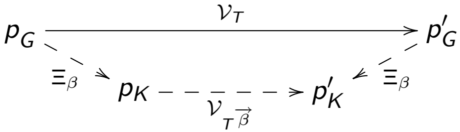


An Example of Ambivalence



Variational Constraints Theorem

- Given
  - Coarse-graining $\beta : G \longrightarrow K$
  - And transmission function T
- Such that T is ambivalent under β
- For any population p_G



Schema Partitionings

- Let G be a set of fixed length bitstrings
- Schema partitioning: A function that maps any genotype to its values at some fixed set of locii.
  - Example: $\beta_{1,3}$ maps any genotype to its 1^{st} and 3^{rd} bits
- Schema partitionings induce schema partitions on the genotype set
- Example continuation: let G be the set of bitstrings of length 8
  - Then $\beta_{1,3}$ induces the schema partition #*#***** over G

Schema Partitionings

- Let G be a set of fixed length bitstrings
- Schema partitioning: A function that maps any genotype to its values at some fixed set of locii.
  - Example: $\beta_{1,3}$ maps any genotype to its 1^{st} and 3^{rd} bits
- Schema partitionings induce schema partitions on the genotype set
- Example continuation: let G be the set of bitstrings of length 8
  - Then $\beta_{1,3}$ induces the schema partition #*#***** over G

Schema Partitionings

- Let G be a set of fixed length bitstrings
- Schema partitioning: A function that maps any genotype to its values at some fixed set of locii.
  - Example: $\beta_{1,3}$ maps any genotype to its 1^{st} and 3^{rd} bits
- Schema partitionings induce schema partitions on the genotype set
- Example continuation: let G be the set of bitstrings of length 8
  - Then $\beta_{1,3}$ induces the schema partition #*#***** over G

Schema Partitionings

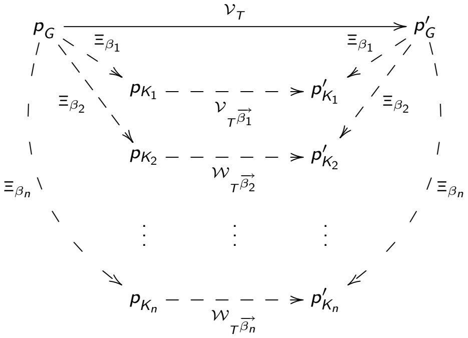
- Let G be a set of fixed length bitstrings
- Schema partitioning: A function that maps any genotype to its values at some fixed set of locii.
  - Example: $\beta_{1,3}$ maps any genotype to its 1^{st} and 3^{rd} bits
- Schema partitionings induce schema partitions on the genotype set
- Example continuation: let G be the set of bitstrings of length 8
  - Then $\beta_{1,3}$ induces the schema partition #*#***** over G

GA Variation is Ambivalent Under any Schema Partitioning

- If genotypes are bitstrings of fixed length
- If the variation operation consists of some combination of
  - n-point crossover
  - Uniform crossover
  - Canonical mutation
    - Probability of mutation is constant for each bit
- Variation is ambivalent under any schema partitioning
- Any schema partitioning can be used to derive a variational constraint

Simultaneous Constraints for GA Variation Operations Using Schema Partitionings

- For any schema partitionings $\beta_1, \ldots, \beta_n$, and any common GA variation operators represented by T



Reduction to a Framework for Studying the Effect of One Evolutionary step on Population Marginals

- In a common GA, *any* schema partitioning β gives us both a selectional constraint and a variational constraint
- 'Concatenating' these constraints we obtain

$$
\begin{array}{ccccc}
p_G & \xrightarrow{\mathcal{S}_f} & p'_G & \xrightarrow{\mathcal{V}_T} & p''_G \\
\downarrow{\Xi_\beta} & & & & \downarrow{\Xi_\beta} \\
p_K & \dashrightarrow_{\mathcal{S}_{F_{(\beta, p_G)}}} & p'_K & \dashrightarrow_{\mathcal{V}_{\overrightarrow{T\beta}}} & p''_K
\end{array}
$$

- Projection of a population through a schema partitioning is essentially a marginalization operation
- Thus for a GA with common variation operations the general framework reduces to a specific framework for studying the effect of one evolutionary step on population marginals

Reduction to a Framework for Studying the Effect of One Evolutionary step on Population Marginals

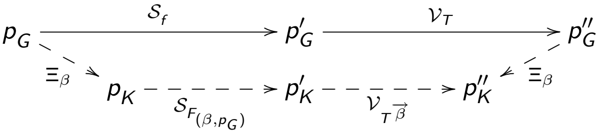
- In a common GA, *any* schema partitioning β gives us both a selectional constraint and a variational constraint
- ‘Concatenating’ these constraints we obtain

$$\begin{array}{ccccccc} p_G & & \xrightarrow{\mathcal{S}_f} & p'_G & \xrightarrow{\mathcal{V}_T} & & p''_G \\ & \searrow^{\Xi_\beta} & & & & \swarrow^{\Xi_\beta} & \\ & p_K & \dashrightarrow_{\mathcal{S}_{F_{(\beta, p_G)}}} & p'_K & \dashrightarrow_{\mathcal{V}_{T\overrightarrow{\beta}}} & p''_K & \end{array}$$

- Projection of a population through a schema partitioning is essentially a marginalization operation
- Thus for a GA with common variation operations the general framework reduces to a specific framework for studying the effect of one evolutionary step on population marginals

Reduction to a Framework for Studying the Effect of One Evolutionary step on Population Marginals

- In a common GA, *any* schema partitioning β gives us both a selectional constraint and a variational constraint
- 'Concatenating' these constraints we obtain



- Projection of a population through a schema partitioning is essentially a marginalization operation
- Thus for a GA with common variation operations the general framework reduces to a specific framework for studying the effect of one evolutionary step on population marginals