Advanced Tutorial on Coevolution

GECCO 2006

Edwin de Jong, Kenneth O. Stanley, and R. Paul Wiegand

2006 Genetic and Evolutionary Computation Conference

Outline

Problem statement I: Optimizing functions over discrete spaces

- **Search space:** A complete solution x consists of n components: $x = x_0, x_1, \ldots, x_{n-1}$
- **Evaluation:** A static fitness function for complete solutions $f(x)$ is given
- **Goal:** Find a complete solution that satifies a property based on f
  Typically: find a complete solution that maximizes f
- **Examples of components:**
  - Partial neural networks
    E.g. a set of neurons, connections, and their weights
  - Bits
  - Note: if the components are bits, we have a regular GA problem

Methods for problems of type I

- Genetic Algorithms
  - Individuals x are complete solutions
  - Evaluation is independent of other individuals $\longrightarrow$ the search process is stable
  - Difficulty: how can good combinations of components be found?
- Cooperative Coevolution
  - Individuals x_i are *components*, i.e. partial specifications
  - To evaluate an individual, the individual is combined with other individuals to form a complete solution
  - Thus, evaluation of an individual is dependent on other evolving individuals $\rightarrow$ the search process may be unstable
  - Example of a cooperative coevolution method:
    given $n - 1$ components, what is the best choice for component i?
    - Optimize components in parallel
    - Optimize components sequentially

Problem statement II: Test-based problems

- **Search space:** A set S of complete solutions.
- **Evaluation:** A second search space T of tests is given. The quality of a solution $s \in S$ is determined by its performance against all tests $t \in T$.
- **Goal:** Find a solution $s \in S$ that maximizes a performance measure based on outcomes against $t \in T$
- **Example:** Chess: find a first-player strategy that has a maximum expected outcome against a randomly chosen opponent t.

Methods for problems of type II

- Evaluate solutions against all tests
  - Generally infeasible
- Evaluate solutions against a random sample of tests
  - Sample may not be representative
  - Finding high quality opponents may be a difficult search problem in itself
- Coevolve the set of tests
  - Secondary search process may identify high quality opponents
  - Potential for open-ended progress, as both solutions and tests improve
  - Evaluation function develops as part of the search process (cf. natural evolution)

Constructing Reliable Coevolution Methods

- Problem with 'naive' coevolutionary setups:
  - Without special arrangements, dynamic evaluation constitutes a moving target
  - Thus, no reason to assume the search process will converge towards desired solutions
- [Ficici, 2004]: importance of selecting a *solution concept*
- Solution concept: divides the search space into solutions and non-solutions
- Example: in a standard GA problem, the solution concept specifies all (and only) maximum-fitness individuals to be solutions.

Methodology for coevolution

- Step I: given an informal problem specification, select or define a solution concept that specifies which objects qualify as solutions
- Step II: Select or design a coevolutionary algorithm that respects the chosen solution concept

Solution Concepts for Cooperative Coevolution

- Maximize fitness: find individuals $x \in X$ that maximizes fitness $f(x)$.
- Maximize robustness: find individuals x such that each component x_i still forms an appropriate choice when the remaining components are varied.

Solution Concepts: Simultaneous Maximization of All Outcomes

Maximize the outcome over all possible tests simultaneously:

$$SC = \{s \in S | \forall s' \in S : \forall t \in T : G(s, T) \geq G(s', T)\}$$

- For many problems, a single solution that simultaneously maximizes the outcomes of all tests does not exist
- Thus, limited application scope
- Monotonic progress: guaranteed by Rosin's *Covering competitive algorithm* [Rosin, 1997].

Solution Concepts: Pareto-Optimal Set

Pareto-coevolution
[Ficici and Pollack, 2000, Watson and Pollack, 2000]: Opponents may be viewed as *objectives*.
Pareto-dominance:
Solution s_1 *dominates* s_2 if:
$s1 \succ s2 \quad \equiv$
$\forall t \in T : G(s_1, t) \geq G(s_2, t) \ \wedge \ \exists t \in T : G(s_1, t) > G(s_2, t)$
Set of all non-dominated individuals: non-dominated set

$$SC = \{s \in S | \nexists s' \in S : s' \succ s\}$$

- Represents all different ways to trade off the different objectives
- Minimal assumptions
- May be very large

Solution Concepts: Pareto-Optimal Equivalence Set

Pareto-optimal set: may contain many individuals solving the same set of tests.
Pareto-Optimal Equivalence set: remove such duplicate solutions.
For each combination of tests that can be solved, the Pareto-Optimal Equivalence Set contains at least one candidate solution that solves it. Since multiple such sets may exist, we $S4$ is the collection of all such sets:
$SC = \{sc \subseteq S | \forall T' \subseteq T :$
$\exists s \in S : solves(s, T') \implies \exists s' \in sc : solves(s', T')\}$

- Equivalently: set that for each member of the Pareto-front contains an equivalent candidate
- Monotonic progress: guaranteed by the Incremental Pareto-Coevolution Archive (IPCA) [De Jong, 2004a]

Solution Concepts: Nash Equilibrium (I)

Players are distributions over the spaces of solutions and tests.
Nash equilibrium: no player can profitably deviate given the strategies of the other players.
Mixed-strategy Nash equilibrium:
n classes of individuals: $I_1, I_2, \ldots I_n$
E.g. $I_1 = S$ and $I_2 = T$. Let $I = \times_{j \in N} I_j$, with $N = \{1, 2, \ldots, n\}$.
$\Delta(I_i)$: the set of probability distributions over I_i
$\Omega = \times_{j \in N} \Delta(I_j)$.
Mixed strategy profile $\alpha \in \Omega$: probability distribution for each class of individuals.
Expected outcome for the i^{th} class of individuals in a mixed strategy profile:
$E(G_i(\alpha)) = \sum_{a \in I} \prod_{j \in N} \alpha_j(a_j) G_i(a)$, where $G_i(a)$ returns the outcome for the i^{th} individual.

Solution Concepts: Nash Equilibrium (II)

A mixed-strategy $\alpha*$ is a Nash-equilibrium if:
$SC = \{\alpha* \in \Omega | \forall i : \forall \alpha_i \in \Delta(I_i) : E(G_i(\alpha*)) \geq E(G_i(\alpha*_1, .., \alpha*_{i-1}, \alpha_i, \alpha*_{i+1}, .., \alpha*_N))))$

- General, game-theoretic solution concept
- Can specify a relatively small set of individuals
- But: there can be (infinitely) many Nash equilibria, part of which may be dominated
- Finding a Nash equilibrium does not guarantee that the highest possible outcomes.
- Monotonic progress: guaranteed by Ficici's *Nash Memory* [Ficici and Pollack, 2003]

Solution Concepts: Maximization of Expected Outcome

Maximize the expected score against a randomly selected opponent:

$$SC = \{s \in S | \forall s' \in S : E(G(s,t)) \geq E(G(s',t))$$

where E is the expectation operator and t is randomly drawn from T.

- Appropriate for many problems, e.g. identifying the best chess player.
- Equivalent to maximizing the sum of an individual's outcomes over all tests, or to a uniform linear weighting of the objectives.
- Assumes all tests are equally important
- Monotonic progress: guaranteed by the MaxSolve algorithm

Pareto-coevolution: informativeness

- Ficici [Ficici and Pollack, 2001]: a test t is informative if it assigns different outcomes to solutions s, s': $G(s,t) > G(s',t)$.
- By using an informative set of tests, accurate evaluation is achieved [Bucci and Pollack, 2002].
- Pareto-coevolution: solution s dominates s' if and only if: $\exists t \in T : G(s,t) > G(s',t)$ and $\nexists t \in T : G(s',t) > G(s,t)$.
- Thus, the *only* required information to determine dominance between s and s' is whether a test exists that makes a distinction between them.
- Therefore, given a set of n solutions, a set TS of at most $n^2 - n$ tests is guaranteed to exist such that evaluation using TS as objectives is equivalent to using *all* tests T as objectives [De Jong and Pollack, 2004].

Underlying Objectives

- The set of all tests specifies a complete set of objectives
- However, for many practical problems, similar tests may test on similar aspects.
- Example: two devices that test whether a bridge can stand forces greater than 8.000 and 10.000 kg respectively.
- Such similar tests can be combined onto a single objective, thus reducing the dimensionality of the evaluation space
- The *underlying objectives* [Bucci et al., 2004, De Jong and Pollack, 2004] of a problem are a minimal set of objectives that provide evaluation equivalent to the set of all tests T.

Coevolutionary pathologies revisited

- Overspecialization: solutions improve on a subset of the underlying objectives
- Disengagement: for one or more underlying objectives, tests are too far apart from solutions to provide a gradient, and thus insufficiently informative
- Intransitivity: by viewing opponents as objectives, rather than as other solutions, any intransitive relations are transformed into transitive ones [Bucci and Pollack, 2003a, De Jong, 2004b].

Introduction to Evolutionary Game Theory

- Evolutionary Game Theoretic (EGT):
  - Tracking *population state* through time
  - Dynamical systems model
  - Discrete time system (map)
  - Interested in properties of limit behaviors
- Model assumptions & properties
  - Population(s): single population, two-population
  - Payoff Properties: reward symmetry, role symmetry
  - Populations: infinite populations, finite populations
  - Interactions: *typically complete mixing*
  - Variation: *typically* none
  - Selection: *typically* proportionate selection
  - Updating: parallel update

Mean of all pair-wise interactions

Modeling the CCEA

- Populations are ratios of genotypes
  - Given n distinct genotypes in a population,
  - $\vec{x} \in \mathbb{R}^n,\ x_i \in [0,1]\ \wedge\ \sum_{i=1}^{n} x_i = 1$

For example: $\vec{x} = \langle 0.2\ 0.1\ 0.7 \rangle$

- Fitness modeled using a *payoff matrix*
  - Treat A a the payoff matrix for a *stage game*
  - a_{ij} is the reward player 1 gets when playing strategy i against player 2's j strategy
  - Strategies in game correspond with evolving *individuals*
- Replicator equations generate the next system state:
  - **Fitness**: The fitness of all strategies in a population is assessed (typically by playing all possible strategies for the other player)
  - **Selection**: The population state vector is updated using a selection method of some sort (typically a proportionate selection)

Terminology

Pure Strategy — A single strategy available to a player in a game

Polymorphic (mixed) Strategy — A *distribution* of pure strategies

Evolutionary Stable Strategy (ESS) — a strategy that, if adopted by a population, cannot be invaded by any alternative strategy

Nash Equilibrium — a strategy set of players in a game with the property that, if all players are playing one of the strategies, no individual player has anything to gain by deviating from their strategy

---

Fixed Point (f.p.) — A point that maps to itself, $\vec{x} = f(\vec{x})$

Stable Fixed Point — A fixed point with the property that all points *near* it stay near it

Basin of Attraction (BOA) — Set of initial conditions that will eventually map to some limit behavior (f.p., cycle, etc.)

Single Population, Test-based Coevolution [Ficici and Pollack, 2000]

- One population, but two players
- Player 1 represents candidate solutions to the problem
- Player 2 represents tests to challenge the solution
- Individuals serve both as strategies for players 1 & 2 (role symmetry)

Fitness:	$\mathcal{F}(\vec{x})$	$\vec{u} = A\vec{x}$
Selection:	$\mathcal{S}(\mathcal{F}(\vec{x}), \vec{x})$	$x_i' = \frac{u_i}{\vec{u} \cdot \vec{x}} \cdot x_i$

- *Population* may represent polymorphic solutions
- Simple CEA cannot recognize such solutions
- CEA can be lead astray by *search constrained* attractive Nash eq.
- Different dynamics can result from two-population algorithms operating on same payoff matrix
- Different dynamics can result if different selection methods are used

Two-Population, Compositional Coevolution [Wiegand et al., 2003]

- Two population, two players
- Player 1 represents candidate for the 1^{st} component of the solution
- Player 2 represents candidate for the 2^{nd} component of the solution
- $f(i, j)$, whether evaluating for player 1 or 2 (reward symmetry)

Fitness:	$\mathcal{F}_x(\vec{x}, \vec{y})$ $\mathcal{F}_y(\vec{x}, \vec{y})$	$\vec{u} = A\vec{y}$ $\vec{w} = A^T\vec{x}$
Selection:	$\mathcal{S}(\mathcal{F}_x(\vec{x}, \vec{y}), \vec{x})$ $\mathcal{S}(\mathcal{F}_y(\vec{x}, \vec{y}), \vec{y})$	$x_i' = \frac{u_i}{\vec{u} \cdot \vec{x}} \cdot x_i$ $y_j' = \frac{w_j}{\vec{w} \cdot \vec{y}} \cdot y_j$

- Nash equilibria are stable, attracting fixed points
- Non-optimal stable f.p. can attract many, most, or all trajectories
- Validation studies suggest that the size of BOAs associated with basis vector f.p. increase as cumulative column/row increase

Applicability of EGT on Finite Population Systems

Not Applicable [Fogel et al., 1995]:

- Pick a simple, two-strategy problem (e.g., Hawk-Dove)
- Pick a simple, finite population CEA
- Does the real CEA converge to the EGT-predicted ESS?
- **Conclusion**: No. Even for very large populations, quantization problems and stochastic noise force the system to deviate from predictions

Applicable [Ficici et al., 2000]:

- Be careful to model the algorithm properly *(if you implement truncation selection, model truncation selection)*
- Use the correct predictions *(e.g., predictions for model adjusted for no self-play)*
- There are modest adjustments to implementation that provide better predictive correlation *(e.g., Baker's SUS rather than proportionate selection)*
- **Conclusion**: Yes. Properly modeled, EGT can be predictive

Finite vs. Infinite Population Models [Liekens et al., 2004]

Some initial observations:

- Infinite populations simplifies analysis *(populations represented simply, models are deterministic)*
- Finite populations complicates things *(use Markov methods, consider all possible populations & compute fixed-point distributions)*
- Prevailing wisdom: predictions of large population models approximate predictions of infinite model
- Reality: Drift can be a powerful factor of finite populations

Finite models behave differently:

- Construct infinite population model using traditional methods (Vose, 1999)
- Construct finite population-genetics based Markov model (Fischer, 1930)
- Include proportionate selection & bit-flip mutation in both models
- Consider three simple 2×2 games: Neutral game, Hawk-Dove, & Prisoner's Dilemma (two pop.)
- Analysis by model iteration
- **Conclusion**: In all cases, small pop. sizes translate to very different behavior from the infinite models

Introduction · Foundational Concepts · Analyzing CEAs · Representation in Coevolution · Conclusion

Coevolutionary Convergence [Schmitt, 2003]

Markov process with certain algorithmic constraints:

- Fully-positive mutation matrix
- Bounded mutation and crossover rate annealing schedule (power-law scaling, with logarithmic exponent)
- Power-law scaled proportionate selection
- $\exists$ strategy set with strictly maximal fitness (i.e., is strictly superior as measured by the other population)

Conclusions:

- A wide variety of (properly scaled versions of) commonly used operators are included in this analysis
- Populations in such CEAs converge asymptotically to a global optima
- Very large populations allow for slow annealing schedules

Introduction · Foundational Concepts · Analyzing CEAs · Representation in Coevolution · Conclusion

Decomposing Solutions for Representation

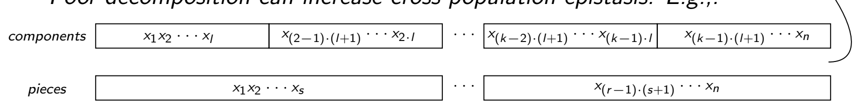
Separability $\sim$ problem's true decomposition

- r inseparable pieces
- Each piece of length s
- $n := rs$
- Function is a linear sum of subfunctions on r pieces

Representing candidate solutions

- Problem decomposition (*piece*)
- Representational decomposition (*component*)
- Two kinds of *representational bias*
  1. Decompositional bias
  2. Linkage bias

Poor decomposition can increase cross-population epistasis. E.g.,:



Introduction · Foundational Concepts · Analyzing CEAs · Representation in Coevolution · Conclusion

Empirical Analysis of Repr. Bias [Wiegand et al., 2002a]

Traditional View:

- High CP epistasis $\rightarrow$ poor coevolutionary performance
- Low CP epistasis $\rightarrow$ good coevolutionary performance
- Compensate by using more complicated interaction methods

Thought Exercise:

- Problem separability aligns perfectly with representational decomposition
- Select *any* collaborator, or just an arbitrary fixed value
- **Observation**: Coevolution is unnecessary!

This is a *bad* example of coevolutionary success

Understanding Repr. Bias:

- Construct problems with different non-linear properties
- Use a *mask* to adjust linkage & decompositional bias
- Consider a variety of collaboration methods
- **Conclusion**: It isn't the *existence* of cross-population epistasis that makes things hard, but the *type*

Partitioning & Focusing

[Jansen and Wiegand, 2004]

Two key aspects of CCEAs:

- Partitioning (separability is important)
- Focusing (increased exploration is important)

Consider 2 simple algorithms:

- (1+1) EA
- CC (1+1) EA

Asymptotic run time analysis:

- Randomized algorithm analysis
- Det. *expected* # evals to max
- Bound probabilities

Consider problems w/ different properties:

- Separable across pop. boundaries
- Inseparable across pop. boundaries

Conclusions:

- Separability insufficient for CCEA advantage
- Separability unnecessary for CCEA advantage
- Inseparability insufficient for EA advantage
- Problem must require *both* partitioning & focusing

Empirical Studies of Methods of Interactions

Methods of Interaction:

- Evaluation in coevolution requires interaction
- Many ways to select competitors / partners
- More interactions per eval → more information, less efficiency
- Less interactions per eval → less information, more efficiency

Some Example Studies:

- [Angeline and Pollack, 1993] Empirical study of different topologies of competitive tournaments
- [Bull, 1997] Broad empirical survey study of performance of partner selection
- [Wiegand et al., 2001] Empirical study of certain properties of collaborator selection
- [Bull, 2001] Formalism for understanding partner selection
- [Panait and Luke, 2002] Broad empirical survey study of performance of competitive evaluations

Conceptualizing the Information Content of Problems

Coevolutionary Problems:

- Coevolutionary problems involve certain structures
- E.g., underlying objectives, dimensions, etc.

Formalisms for Studying Problem Structure:

- [Rosin and Belew, 1997] *teaching set* — set of individuals capable of defeating all possible nonoptimal opponent
- [Ficici and Pollack, 2001] *distinction* — If learner x performs better than learner y with respect to teacher j, we say that teacher j *distinguishes* the learner pair (x, y) in favor of x
- [Bucci and Pollack, 2003b] *maximally informative test set* — the set of tests having neither incomparable elements nor equal elements
- [De Jong and Pollack, 2003] *complete evaluation set* — set of individuals capable to detecting all selectable differences between learners

Problem Classes for Analysis

Constructing Problem Classes:

- Tunable: A range of problem instances can be generated straightforwardly
- Demonstrative: Illustrate certain problem properties
- Simple: Preferably, analytically tractable in some way
- Challenging: Possible to generate instances difficult for coevolution

Some Example Problem Classes:

- [Kauffman and Johnsen, 1992] Probabilistic, coupled landscapes: NKC
- [Watson and Pollack, 2001] Minimal substrate: Numbers games
- [Wiegand et al., 2002b] Tunable miscoordination: MaxOfTwoQuadratics
- [Popovici and De Jong, 2006] Tunable best response: ridge / plateau functions

Measuring Coevolution

Diagnosing CEA Behavior:

- Red Queen dynamics & poor solution concept formulation make it hard to discern coevolutionary progress
- Coevolution generates many pathological behaviors

Some Example Measures:

- [Cliff and Miller, 1995] Current individual vs. ancestral opponent
- [Pollack and Ficici, 1998] Order statistics and measured entropy
- [Stanley and Miikkulainen, 2002a] Dominance tournament
- [Bader-Natal and Pollack, 2004] All of generation visualization

Understanding Performance

Best Response:

- Best-responses are a problem property
- Trajectories of best individuals through the search space (algorithm property) tend to approximate the best responses
- Accuracy of approximation depends on CEA parameter settings
- High accuracy bad when ∃ multiple nash equilibria (intersections of best-responses) of different values

Issues Analyzed:

- Competitions [Popovici and De Jong, 2004]
- Collaboration methods in compositional coevolution [Popovici and De Jong, 2005a]
- Population sizes & elitism in compositional coevolution [Popovici and De Jong, 2005b]

Random Walk Theory:

- Consider a simple CEA
- On variations of the numbers game
- Compare behavior to random walk

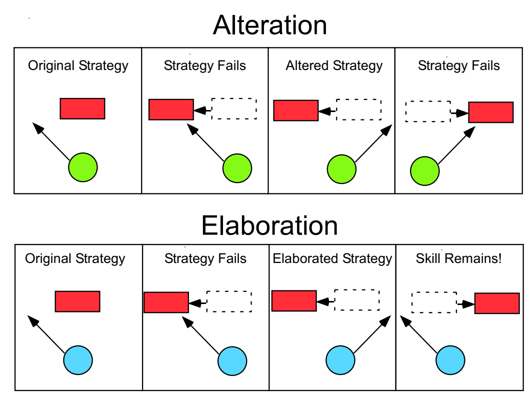
Issues Analyzed:

- Intransitivity [Funes and Pujals, 2005]

Choosing Opponents Is Not the Only Problem

- How can new solutions be continually created that maintain existing capabilities?
- Mutations that lead to innovations could simultaneously lead to losses
- What kind of process ensures elaboration over alteration?

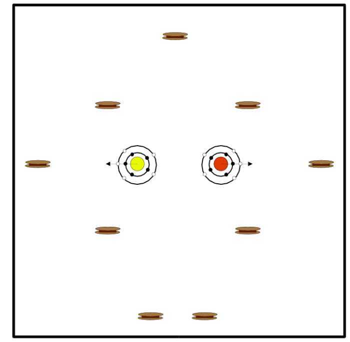
Alteration vs. Elaboration



Encoding Affects Performance

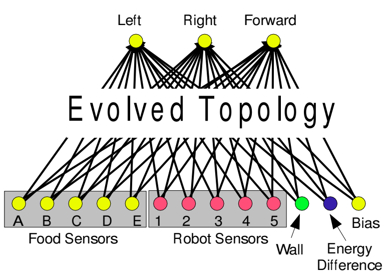
- Fixed length genomes limit progress
- Dominant strategies that utilize the entire genome must alter and thereby sacrifice prior functionality
- If new genes can be added, dominant strategies can be elaborated, maintaining existing capabilities
- → Complexification is an important process for the encoding

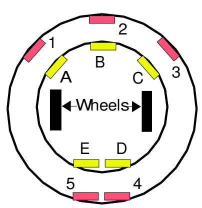
Example Domain: Robot Duel



- Robot with higher energy wins by colliding with opponent
- Moving costs energy
- Collecting food replenishes energy
- Complex task: When to forage/save energy, avoid/pursue?

Robot Neural Networks



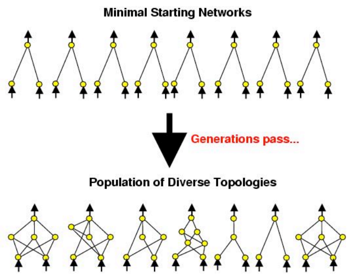


-

Set of Strategies Not Fixed or Known

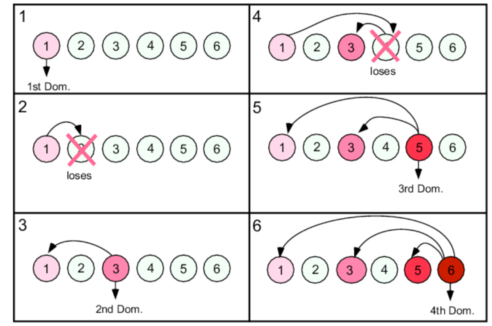
- Progress should continue indefinitely
- Solutions should elaborate
- Should not require estimating task complexity
- → Use a method that complexifies

Complexifying Method: NeuroEvolution of Augmenting Topologies (NEAT)



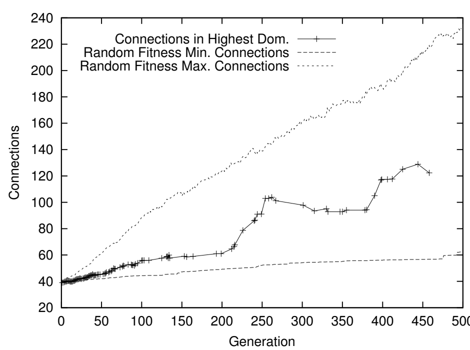
- NEAT evolves increasingly complex neural network for control [Stanley and Miikkulainen, 2004]
- Mutations occasionally add new structure
- Speciation protects innovative structures
- Successful elaborations survive
- → The populaton complexifies

Dominance Tournament Progress Measure [Stanley and Miikkulainen, 2002b]



- The first dominant strategy d_1 is the generation champion of the first generation;
- dominant strategy d_j, where $j > 1$, is a generation champion such that for all $i < j$, d_j is superior to (wins the 288 game

Evolution of Complexity



- As dominance increases so does complexity on average
- Networks with strictly superior strategies are more complex

Comparing Performance

Coevolution Type	**Ave. Highest Dom. Level**	**Average Performance**	**Equivalent Generation (out of 500)**
Complexifying	15.2	91.4%	343
Fixed-Topology 10 Hidden Node	12.0	40.4%	24
Fixed-Topology 5 Hidden Node	13.0	80.3%	159
Fixed-Topology Best Network	14.0	82.4%	193
Simplifying	23.2	57.3%	56

Coevolution in Practice

- Evaluation is expensive
  - Choosing opponents/teammates must involve sampling
  - Even under theoretically-founded schemes
- Solution space is unknown/undefined
  - Representation must be open-ended
  - Genomes need to complexify
- Is coevolution ready for real-world applications?

Introduction | Foundational Concepts | Analyzing CEAs | Representation in Coevolution | Conclusion

Some Final Remarks

- It is important to know what we're solving
  - Invest time in formalizing the solution concept of the problem
  - Try to apply a CEA appropriate for that concept
- Theory is progressing — many tools are now available:
  - Evolutionary game theory & dynamical systems analysis
  - Markov modeling & Markov chain analysis
  - Randomized run-time analysis
  - Order theory & information theoretic approaches
  - A variety of useful problem classes
  - Dynamics analysis (e.g., best-response)
- Practical applications of coevolution may require special considerations
  - Representation is critical
  - Expanding the search space enables continual elaboration

Introduction | Foundational Concepts | Analyzing CEAs | Representation in Coevolution | Conclusion

Angeline, P. J. and Pollack, J. B. (1993).
Competitive environments evolve better solutions for complex tasks.
In *Proceedings of the 5th International Conference on Genetic Algorithms, ICGA-93*, pages 264–270. Morgan Kaufmann.

Bader-Natal, A. and Pollack, J. (2004).
A population-differential method of monitoring success and failure in coevolution.
In *Proceedings from the Genetic and Evolutionary Computation Conference*.

Bucci, A. and Pollack, J. B. (2002).
Order-theoretic analysis of coevolution problems: Coevolutionary statics.
In *Proceedings of the GECCO-02 Workshop on Coevolution: Understanding Coevolution*.

Bucci, A. and Pollack, J. B. (2003a).
Focusing versus intransitivity. Geometrical aspects of coevolution.
In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-03*. Springer.

Bucci, A. and Pollack, J. B. (2003b).
A mathematical framework for the study of coevolution.
In *Foundations of Genetic Algorithms (FOGA-2002)*. Morgan Kaufmann.

Bucci, A., Pollack, J. B., and De Jong, E. D. (2004).
Automated extraction of problem structure.
In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-04*, pages 501–512.

Introduction | Foundational Concepts | Analyzing CEAs | Representation in Coevolution | Conclusion

Bull, L. (1997).
Evolutionary computing in multi-agent environments: Partners.
In *Proceedings of the 5th International Conference on Genetic Algorithms, ICGA-97*, pages 370–377.

Bull, L. (2001).
On coevolutionary genetic algorithms.
Soft Computing, 5(3):201–207.

Cliff, D. and Miller, G. F. (1995).
Tracking the Red Queen: Measurements of adaptive progress in co-evolutionary simulations.
In *Proceedings of the Third European Conference on Artificial Life: Advances in Artificial Life*, volume 929 of *LNAI*, pages 200–218. Springer.

De Jong, E. D. (2004a).
The Incremental Pareto-Coevolution Archive.
In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-04*, pages 525–536.

De Jong, E. D. (2004b).
Intransitivity in coevolution.
In *Parallel Problem Solving from Nature - PPSN VIII*, volume 3242 of *LNCS*, pages 843–851. Springer-Verlag.

De Jong, E. D. and Pollack, J. B. (2003).
Learning the ideal evaluation function.

In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-03*, pages 274–285. Springer.

De Jong, E. D. and Pollack, J. B. (2004).
Ideal evaluation from coevolution.
Evolutionary Computation, 12(2):159–192.

Ficici, S. G. (2004).
Solution Concepts in Coevolutionary Algorithms.
PhD thesis, Brandeis University.

Ficici, S. G., Melnik, O., and Pollack, J. B. (2000).
A game-theoretic investigation of selection methods used in evolutionary algorithms.
In *Proceedings of the 2000 Congress on Evolutionary Computation, CEC-00*. IEEE Press.

Ficici, S. G. and Pollack, J. B. (2000).
A game-theoretic approach to the simple coevolutionary algorithm.
In *Parallel Problem Solving from Nature, PPSN-VI*. Springer.

Ficici, S. G. and Pollack, J. B. (2001).
Pareto optimality in coevolutionary learning.
In *Sixth European Conference on Artificial Life*. Springer.

Ficici, S. G. and Pollack, J. B. (2003).
A game-theoretic memory mechanism for coevolution.

In *Genetic and Evolutionary Computation – GECCO-2003*, volume 2723 of *LNCS*, pages 286–297. Springer-Verlag.

Fogel, D., Fogel, G., and Andrews, P. (1995).
On the instability of evolutionary stable strategies.
BioSystems, 44:135–152.

Funes, P. and Pujals, E. (2005).
Intransitivity revisited coevolutionary dynamics of numbers games.
In *Proceedings from the Genetic and Evolutionary Computation Conference*, pages 515–521.

Jansen, T. and Wiegand, R. P. (2004).
The cooperative coevolutionary (1+1) ea.
Evolutionary Computation, 12(4):405–434.

Kauffman, S. A. and Johnsen, S. (1992).
Co-evolution to the edge of chaos: Coupled fitness landscapes, poised states, and co-evolutionary avalanches.
In *Artificial Life II*, pages 325–369. Addison-Wesley.

Liekens, A., Eikelder, H., and Hilbers, P. (2004).
Predicting genetic drift in 2×2 games.
In *Proceedings from the Genetic and Evolutionary Computation Conference*, pages 549–560.

Panait, L. and Luke, S. (2002).
A comparison of two competitive fitness functions.

In *Proceedings from the Genetic and Evolutionary Computation Conference*, pages 503–511.

Pollack, J. B. and Ficici, S. G. (1998).
Challenges in coevolutionary learning: Arms-race dynamics, open-endedness, and mediocre stable states.
In *Proceedings of the Sixth International Conference on Artificial Life*, pages 238–247. MIT Press.

Popovici, E. and De Jong, K. A. (2004).
Understanding competitive co-evolutionary dynamics via fitness landscapes.
In *Proceedings from the AAAI 2004 Fall Symposium on Artificial Multiagent Learning*. AAAI Press.

Popovici, E. and De Jong, K. A. (2005a).
A dynamical systems analysis of collaboration methods in cooperative co-evolution.
In *Proceedings of the AAAI 2005 Fall Symposium on Coevolutionary and Coadaptive Systems*. AAAI Press.

Popovici, E. and De Jong, K. A. (2005b).
Understanding cooperative co-evolutionary dynamics via simple fitness landscapes.
In *Proceedings of the Genetic and Evolutionary Computation Conference*.

Popovici, E. and De Jong, K. A. (2006).
The dynamics of the best individuals in co-evolution.
Natural Computing, (to appear).

Rosin, C. D. (1997).
Coevolutionary Search among Adversaries.
PhD thesis, University of California.

Rosin, C. D. and Belew, R. K. (1997).
New methods for competitive coevolution.
Evolutionary Computation, 5(1):1–29.

Schmitt, L. (2003).
Coevolutionary convergence to global optima.
Technical report, The University of Aizu, Aizu-Wakamatisu City, Japan.

Stanley, K. O. and Miikkulainen, R. (2002a).
Continual coevolution through complexification.
In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-02:*, pages 113–120. Morgan Kaufmann.

Stanley, K. O. and Miikkulainen, R. (2002b).
The dominance tournament method of monitoring progress in coevolution.
In *GECCO 2002: Proceedings of the Bird of a Feather Workshops, Genetic and Evolutionary Computation Conference*, pages 242–248. AAAI.

Stanley, K. O. and Miikkulainen, R. (2004).
Competitive coevolution through evolutionary complexification.
Journal of Artificial Intelligence Research, 21:63–100.

Watson, R. A. and Pollack, J. B. (2000).

Symbiotic combination as an alternative to sexual recombination in genetic algorithms.
In *Parallel Problem Solving from Nature, PPSN-VI*. Springer.

Watson, R. A. and Pollack, J. B. (2001).
Coevolutionary dynamics in a minimal substrate.
In *Proceedings of the Genetic and Evolutionary Computation Conference, GECCO-01*, pages 702–709. Morgan Kaufmann.

Wiegand, R. P., De Jong, K. A., and Liles, W. C. (2002a).
The effects of representational bias on collaboration methods in cooperative coevolution.
In *Parallel Problem Solving from Nature, PPSN-VII*, pages 257–268.

Wiegand, R. P., Liles, W., and De Jong, K. (2001).
An empirical analysis of collaboration methods in cooperative coevolutionary algorithms.
In *Proceedings from the Genetic and Evolutionary Computation Conference*, pages 1235–1242.

Wiegand, R. P., Liles, W., and De Jong, K. (2002b).
Analyzing cooperative coevolution with evolutionary game theory.
In *Proceedings of the 2002 Congress on Evolutionary Computation CEC2002*, pages 1600–1605. IEEE Press.

Wiegand, R. P., Liles, W. C., and De Jong, K. (2003).
Modeling variation in cooperative coevolution using evolutionary game theory.
In *Foundations of Genetic Algorithms 7*, pages 203–220. Morgan Kaufmann.