ANALYZING IMPERFECTIVE GAMES
(THE COMPANION PAPER)

IGOR YANOVICH

Abstract. [Deo, 2015] is the first study applying mathematically explicit evolutionary analysis to a specific semantic-change phenomenon, namely the progressive-imperfective diachronic cycle. The empirically observed cycle is as follows. It starts with grammar (a) where only one form X is used both for progressive and imperfective meanings. Then in (b), optional progressive marker Y is innovated; in (c), Y becomes the obligatory marker for progressive meanings, while X marks the imperfective. At the final stage (d), old form X dies out altogether, and Y expresses both the imperfective and the progressive. Though reaching (d) from (a), Deo's original model does not predict that either (b) or (c) is learned by all speakers in the community. The present paper discusses how Deo's general framework behaves under a wide range of parameter values, what implicit assumptions about the nature of language change it encodes, and what other assumptions may be used instead. While it is possible to improve on Deo's original predictions, no choice of parameters results exactly in the sequence (a)→(b)→(c)→(d) sequence. Either (b) is never spoken by all speakers, or (d) is not reached, or else back-shifts such as (c)→(b) appear. This evolutionary analysis makes a novel empirical prediction: if Deo's general model is correct, the actual progressive-imperfective cycle should look like one of the predicted alternatives. Further empirical research will either confirm those predictions, or falsify Deo's model of the cycle.

Date: July 28, 2015

The author is very grateful to Ashwini Deo and Roland Mühlenbernd for discussions, and to the Alexander von Humboldt Foundation for its support.

This paper is a longer companion to a short summary paper currently under submission.

Contents

1. Introduction

The empirical focus of [Deo, 2015]'s analysis is the **progressive-imperfective diachronic cycle**. The present paper analyzes and extends the evolutionary game-theoretic part of Deo's analysis. Deo's evolutionary modeling was meant to provide a plausible theory of change to complement Deo's semantic analysis of the progressive-imperfective cycle. There are two reasons to analyze further Deo's evolutionary model. First, [Deo, 2015] only provides a single set of parameters for her model, though, as we will see, parameter choices dramatically affect the predictions. Moreover, in some cases implicit assumptions enter the model, such as in Deo's choice of specific evolutionary dynamics equations. When we examine those assumptions, we will see that they are not intuitively plausible to historical linguists. Overall, examining how Deo's model behaves under a variety of parameter settings will allow us a better appreciation of what exactly it predicts about the progressive-imperfective cycle of change.

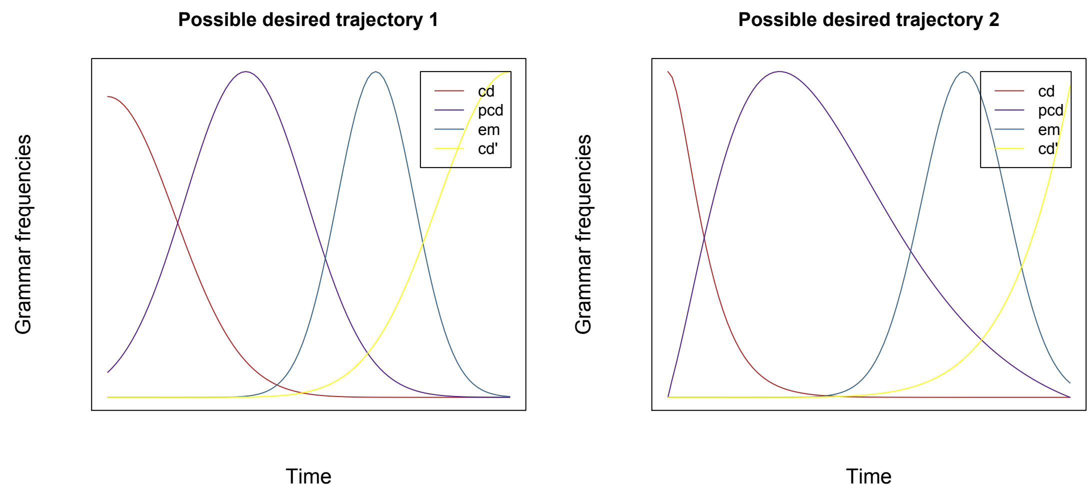
The second reason to study Deo's model further is that as it stands, it fails to derive an important property of the progressive-imperfective cycle. The linguistic analysis of the cycle in various language families leads one to expect trajectories more or less like those in (1). Two things are crucial about them: (i) development eventually leads from the initial stage (red line close to 100% of speakers) to the final stage (yellow line close to 100%); (ii) in the middle, relatively stable intermediate linguistic systems are established (that is, there are moments when the purple and blue lines each reach close to 100%). Deo's actual predictions, in (2), support the feature (i), but not feature (ii).[1] It is thus a natural question if other parameter settings could predict both desiderata (i) and (ii). To give a short preview, we will see that there are principled reasons why Deo's model is actually not capable of deriving *exactly* the trajectories in (1), though we will do better than (2). This

[1] [Deo, 2015] is aware of that, and re-defines the second and third stages in her (43). She considers a population to be in the second stage when the number of S1 (first stage) and S2 (second stage) speakers is greater than that of S3, and in addition the number of S2-speakers is greater than 30%. The choice of the cutoff is likely related to the fact that under Deo's original parameter settings, the share of S2 hardly goes up to 50%.

To determine whether such a redefinition is plausible, we need to consider its linguistic interpretation. S2 is the strategy of having a form that may specifically mark progressive contexts, but is not obligatory to use in such: the simple imperfective form also remains grammatical. Then what Deo's re-definition targets is a population where only 30% of speakers use the optional innovative progressive form. Early Modern English is a good example of S2: the progressive form was developed in Late Middle English (transition from S1 to S2), but it is only in the 19th century that the simple present ceased to be used for description of "progressive events" (transition from S2 to S3), cf. [Seoane, 2012, Sec. 4.2] for overview and references. It is highly doubtful that only, say, 40% of English speakers in 1700 knew how to produce and interpret the progressive form, and everybody else had no idea.

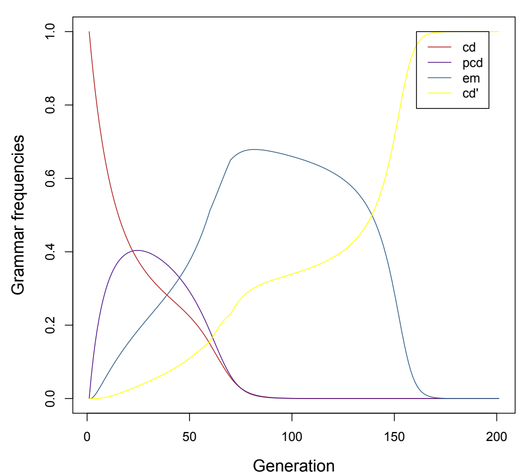
means that either the model is not correctly capturing the phenomenon, or else that the actual change trajectories differ from the expected ones in (1). Whether Deo's model is generally on the right track thus becomes an empirical question.

(1) **Imperfective-progressive trajectories that roughly correspond to empirical observations**



(2) **The evolutionary trajectory predicted by** [Deo, 2015]

(a replica of [Deo, 2015, Fig. 4])



Deo's evolutionary analysis is an innovation in historical semantics, seeking to couple semantic theories of the changing elements with explicit modeling that captures their dynamic behavior. Deo uses her evolutionary analysis essentially as a proof of concept, demonstrating the general plausibility of the proposed theory of the progressive-imperfective cycle. The broader methodological goal of the present paper is to go beyond plausibility checking, and in the course of that to further the case for evolutionary modeling in semantics and historical linguistics. By disassembling Deo's system into components and explaining how they affect the outcome, I aim to make evolutionary modeling more accessible to a practicing linguist. By going over the many cases where the system fails, and stressing the linguistic interpretation of various modeling choices, I hope to demonstrate how modeling can help to derive empirical predictions. Importantly, many of such derived predictions will be wrong, and therefore falsify the assumptions put into the modeling. The present paper thus follows the line of applying evolutionary analysis to falsification of specific theories of language change as exemplified by [Baxter et al., 2009] and [Blythe and Croft, 2012].

The paper proceeds as follows. Section 2 briefly introduces the progressive-imperfective cycle as an empirical phenomenon of language change. Section 3 presents the evolutionary game-theoretic setup of [Deo, 2015]. Section 4 shows how the fitness and mutation components of Deo's system work in isolation, and then together. In particular, we will see how the basic parameters of Deo's system affect the existence and position of evolutionary equilibria. Section 5 demonstrates what happens if we abandon Deo's implicit assumption that the population of speakers is infinite. This change would not just make our model more intuitively plausible, but actually allow us to derive much better predictions. Specifically, in finite populations it becomes possible to derive the establishment of the middle-stage grammars of the progressive-imperfective cycle. In Section 6, I discuss the implicit assumptions that Deo adopts by choosing the replicator-mutator evolutionary dynamics for language change, and propose an alternative linguistic interpretation for her evolutionary dynamics. We will also examine how to turn one's theoretical assumptions about the laws of language change into an explicit evolutionary dynamics. In Section 7, I sketch how an analysis of the progressive-imperfective cycle may look like in the evolutionary framework where individual utterances rather than grammars compete, along the lines of [Croft, 2000]. I conclude that there is no need to select one of these two major approaches as "right": each is better suited for different purposes, and the choice between the two should be guided by convenience. Section 8 concludes, outlining the new insights regarding evolutionary modeling for semantic, and more broadly, grammar change, and stating some open questions.

Importantly, the best trajectories that we can derive building on Deo's semantic and game-theoretic analysis still do not fully match the expected trajectories in (1). But we actually do not know what the real historical trajectories of such semantic developments look like: there are no empirical studies with sufficient temporal resolution. It can be that such future studies will uncover exactly the trajectories that we derived through modeling. On the other hand, the actual trajectories may indeed be different, thus falsifying the model. But the point is that before we engaged with evolutionary models, we did not have the means to make such fine-grained projections to test through empirical studies.

Annotated R code for generating the experiments and the illustrations is provided in supplementary materials. Each diagram in the main part of the paper is accompanied by the code generating either that diagram, or a comparable one in the case of stochastic evolutionary dynamics for which no two runs will be exactly alike. To generate a diagram, one needs first to load the code in the supplied source files into R, by using command `source("YOURPATH/ImpGame_load_all.r")`, which will load all other source files. `YOURPATH` stands for the directory to which you downloaded the R files. Once those files are loaded, the commands for generating a diagram that are provided in the main text can be run.

The source files may also be opened, edited in a text editor, and re-loaded into R. They contain brief annotations explaining each function and parameter. Code has not been optimized for speed. However, individual runs should not impose unreasonable computational load on modern machines.

2. The empirical phenomenon: the progressive-imperfective cycle

Cyclic diachronic developments have been discovered in many areas of language. For example, verbs may lose some of their inflection, while new periphrastic constructions with auxiliary verbs may gradually start to express the meanings of the old inflections. Over time, such auxiliaries may merge with the verbal stem to form new inflections. This is what happened, for example, in the development of Romance futures: (a) Latin had future inflections, as in *cantā-bō* 'I will sing'; (b) a futurate construction "V.INF have" developed already in Late Latin of the form *cantāre habeō* (sing.INF have.1SG) ≈ 'I will sing'; (c) the futurate turned into a single-word, inflectional future form "V-have" in the Romance languages, cf. French *chanterai* ⇐ *chanter* 'sing.INF' + *ai* 'I have'.[2] The end point of this development may feed a new cycle: e.g., the modern Romance futures may be lost at some point. In fact, the Latin form *cantābō* 'I will sing' itself derives from a combination of *canta* 'sing.NOUN' with Indo-European verb *bhwō* 'to be.SUBJUNCTIVE', [Ernout, 1914, pp. 228-30].[3]

As another example, famously known as the Jespersen cycle after [Jespersen, 1917], emphatic words may start to be associated specifically with negative contexts, then lose their literal meaning and become a part of the negative grammatical pattern, later lose their emphatic quality, and finally become the neutral way to express negation, with the older negation fading away from the language. Thus in modern

[2]This is a somewhat simplified version of the Romance story. In particular, not all modern Romance developed the future out of combinations with the "have" verb.

[3]For overview, see e.g. [Fleischman, 1982].

literary French, negation has two obligatory parts *ne ... pas*, where *ne* is the old negation, and *pas* historically derives from the word with literal meaning 'step'. That 'step' word could appear emphatically in negative contexts already in Old French, but is now the actual exponent of negation: In vernacular French, the *ne* part of the *ne ... pas* pattern is no longer obligatory, and the negative force is carried by *pas* (cf. [Eckardt, 2006, Ch. 5] for a fine-grained semantic analysis of the process.) Again, the situation in the modern vernacular French may feed a new instance of the Jespersen cycle later.

The progressive-imperfective cycle is another such cycle. It works as follows.[4] First, the language in question has a single verbal form X_{imp} which may express a wide range of meanings, including "narrow-imperfective" ones as in "Mary has a sister" and progressive ones as in "Mary is riding a bike" (Deo calls this stage *cd*, from *c(ontext) d(ependepnt)*, as hearers need to resolve the single form based on the context.) Second, a new construction emerges that is specialized for expressing progressive meanings, Y_{prog}. The old form X_{imp} may still be used in progressive contexts, though. (This is stage *pcd*, for *p(artially) c(ontext) d(ependepnt)*: for X_{imp}, the hearers still have to resort to context, while Y_{prog} unambiguously signals a progressive meaning.) At the third stage, X_{imp} and Y_{prog} become complementarily distributed, with X_{imp} restricted to narrow-imperfective contexts, and Y_{prog} the only way to express progressive meanings. (This is stage *em*, for *e(xplicit) m(arking)*.) Finally, Y_{prog} undergoes semantic generalization and starts being used in narrow-imperfective contexts, too, thus becoming Y_{imp}. Eventually the old X_{imp} disappears from the language completely, with only Y_{imp} left. (Stage cd', repeating the initial stage *cd*, but with the new form Y in place of X.) Schematically, this is shown in the diagram (3).

(3) **The imperfective-progressive cycle:**

$$\begin{tikzcd} \boxed{pcd\text{: X}_{imp}, (\text{Y}_{prog})} \arrow[r, Rightarrow] & \boxed{em\text{: X}_{imp}, \text{Y}_{prog}} \arrow[d, Rightarrow] \\ \boxed{cd\text{: X}_{imp}} \arrow[u, Rightarrow] & \boxed{cd'\text{: Y}_{imp}} \end{tikzcd}$$

3. Deo's evolutionary modeling: the Imperfective Game

[Deo, 2015]'s analysis of the progressive-imperfective cycle consists of a semantic and an evolutionary parts. The semantic part provides formal analysis and functional

[4]We follow [Deo, 2015]'s characterization, which in turn follows [Bybee et al., 1994, Ch. 5].

justification for the individual changes constituting the imperfective-progressive cycle. It proposes what exactly the semantic systems at each stage are like, and what the speakers participating in the cycle are actually doing semantically when the old system develops into a new one. The evolutionary part attempts to derive the desired trajectory of change on the basis of the semantic part of the analysis. The semantic and the evolutionary parts work in tandem: they are both needed to derive any predictions about the change trajectory. In this paper, we are only concerned with the evolutionary analysis. In what follows, I presuppose the semantic part of [Deo, 2015] without discussion.

3.1. **The basic Imperfective Game and the Normal-Form Imperfective Game.** There are several analytic steps that we need to take to arrive from the semantic analysis of [Deo, 2015] to evolutionary predictions: the basic Imperfective Game; the Normal-Form Imperfective Game (NF ImpGame); the Evolutionary Imperfective Game (Ev ImpGame). Deo's basic Imperfective Game models interactions that employ imperfective and progressive markers between speakers with possibly different grammars. This game gives rise to the Normal-Form Imperfective Game, which encapsulates the analysis of individual interactions in a simple matrix of payoffs for speakers with particular grammars. Finally, the Normal-Form Imperfective Game serves as the underlying game of the Evolutionary Imperfective Game that represents the long-term dynamics of imperfective-progressive interactions in a large population of agents. Going from NF ImpGame to Ev ImpGame, we need to make a large number of design choices. The core of the present paper shows how different our predictions will be based on those choices. But before we get to that, we need to describe the underlying games.

The basic Imperfective Game is played between two players, speaker and hearer. The analytic purpose of the game is to spell out the degree of communicative success of different progressive-imperfective grammars when their bearers interact. In other words, the games captures the functional considerations, based on efficiency of information transfer, that are supposed to direct the process of language chance.

There are two nature states in the game which the speaker may want to convey to the hearer, **phen** and **struc**. **phen** models real-world situations where a progressive meaning is appropriate; **struc**, those where a narrow-imperfective meaning is. The speaker observes a nature state, and needs to transmit it to the hearer who does not observe it directly. The hearer then guesses the state, taking into account the speaker's signal. If the hearer guesses correctly, both players get a payoff of 1, i.e. the round of play was communicatively successful. Otherwise, both receive 0, i.e. we have communicative failure.

Each message is sent in one of two types of (extralinguistic) context, C_{phen} and C_{struc}. In C_{phen}, nature state **phen** occurs with probability 0.9, and nature state

struc with probability 0.1. It is the other way round for C_{struc}. Unlike nature states, contexts are observed by both speaker and hearer. Thus the hearer, when guessing the speaker's intention, can use the information about the current context as well as the linguistic message. C_{phen} and C_{struc} are equally likely to occur.

There are two signals speakers may in principle use, X_{imp} and Y_{prog}. There are four strategies available to speakers, (4). These are intended to model precisely the types of empirically observed grammatical systems with progressive and imperfective forms. There are also four deterministic hearer strategies, (5), naturally corresponding to the speaker strategies. The full strategy of a speaker-hearer is formed by a pair of an S and an H strategy. Only four "natural" pairs are considered: $\langle S_{cd}, H_{cd}\rangle$, $\langle S_{pcd}, H_{pcd}\rangle$, $\langle S_{em}, H_{em}\rangle$, $\langle S_{cd'}, H_{cd}\rangle$. The underlying idea here is that each speaker-hearer is in possession of a *grammar* that deterministically regulates their choice of X_{imp} and Y_{prog}. Their S and H strategies are determined by that grammar, and it is presupposed that the "speaking grammar" and the "hearing grammar" are identical. [5] We will henceforth use *cd* as shortcut for $\langle S_{cd}, H_{cd}\rangle$, and so on.

(4) **Speaker strategies in the Imperfective Game:**

(i) S_{cd}, from "context dependent": only X_{imp} is used, so the only information that helps the hearer to recover **phen** or **struc** is from the prior on how frequently they occur in C_{phen} and C_{struc}.

(ii) S_{pcd}, from "partially context dependent": in **struc**, speakers always choose X_{imp}; in **phen**, they choose X_{imp} in C_{phen}, but Y_{prog} in C_{struc}. In other words, when the context disfavors **phen**, speakers select the form Y_{prog} that specifically signals that the transmitted meaning is that of a progressive.

(iii) S_{em}, from "explicit marking": speakers always choose X_{imp} in **struc**, and Y_{prog} in **phen**.

[5] Generally, this need not be the case in language change. Sociolinguistic research has uncovered the systematic phenomenon called *near-merger* when speakers may consistently produce a linguistic distinction, but nevertheless cannot distinguish the difference in perception. To my knowledge, this has only been shown for phonological distinctions, where state-of-the-art data-collection procedures allow to reach such conclusions with certainty. Demonstrating similar phenomena for morphological, syntactic or semantic distinctions with the same degree of rigor would likely require novel methodologies. An extensive discussion of near-mergers and methods for detecting them may be found in [Labov, 1994, Ch. 10-14]. What complicates the picture even further is that (a) speakers may vary with respect to how well they reproduce and how well they can perceive the near-merged distinction, and (b) production of the distinction may vary across more careful vs. less careful speech styles by the same individual (more careful styles will exhibit merged productions, while in spontaneous speech they would diverge).

(iv) $S_{cd'}$, same as *cd*, only with Y_{prog} as the only form.

(5) **Hearer strategies in the Imperfective Game:**

(i) H_{cd}: $C_{phen} \rightarrow$ **phen**, $C_{struc} \rightarrow$ **struc**

(ii) H_{pcd}: $C_{phen} \rightarrow$ **phen**, $\langle C_{struc}, X_{imp}\rangle \rightarrow$ **struc**, $\langle C_{struc}, Y_{prog}\rangle \rightarrow$ **phen**

(iii) H_{em}: $X_{imp} \rightarrow$ **struc**, $Y_{prog} \rightarrow$ **phen**

(iv) $H_{cd'} = H_{cd}$

In strategy *cd*, agents only use the old form X_{imp}, and try to recover the message **phen** or **struc** based on their prior expectations about the current context: e.g., in C_{phen}, hearers assume the speaker meant **phen**. As contexts are good predictors for the message in 90% cases, this strategy leads to the overall efficiency of transfer at exactly 90% on average. In strategy *pcd*, speakers use a special innovative form Y_{prog} when they are in C_{struc}, but want to signal **phen**. This leads to an even higher efficiency of communication. In strategy *em*, agents disregard the context, and signal **phen** with Y_{prog} and **struc** with X_{imp}. Finally, in cd', the old form X is lost, and the newer form Y is the only signal the agents use. They recover **phen** and **struc** on the basis of extralinguistic context alone, and we are back to the initial stage of the cycle.

Note that the choice of available grammars for the players implies a more specific view of how the progressive-imperfective cycle works than the general understanding in the historical-linguistic literature directly supports. First, it is not known whether at the stage where the progressive is still optional, speakers use it only, or at least dominantly, when the extralinguistic context favors an imperfective meaning. Of course, it would be plausible if speakers did; but at the moment we do not have hard data to either support or refute that. Second, and perhaps more importantly, as [Bybee et al., 1994] observe and [Deo, 2015] repeats, there exist languages that use old X and new Y forms interchangeably. Later on, such systems presumably give way to single-form Y systems, with X dying out. But in the basic Imperfective Game, such stage is not defined. The only options are categorical grammar *em* and only-Y grammar cd'.

The strategies *cd* and cd' employ only one linguistic form, while *pcd* and *em* employ two. Presumably, it is more costly to maintain a linguistic system with more forms, other things being equal. Therefore we introduce the cost parameter k which punishes the two-form strategies. The punishment is formalized as a penalty k included into the payoff in each round for a speaker who uses strategy *pcd* or *em*.[6]

[6]One could argue that for the hearer, maintaining a two-form system is also costly, though [Deo, 2015] only punishes the speaker. Fortunately, this need not concern us: if we add the same

For concreteness, here is how one round of the basic ImpGame may proceed. The speaker has strategy *pcd*, the hearer has the strategy *cd*. The context is C_{struc}. The nature selects state **phen** (though it's only 0.1-likely to arise in C_{struc}, it's still a possibility.) Now the speaker looks up her strategy to determine what signal she should send. It is Y_{prog}. The hearer receives Y_{prog}, and notes that the context is C_{struc}. According to her strategy *cd*, she disregards the speaker's message, and guesses that the nature state was **struc**. As it happens, she guessed wrongly in this case, though in 90% of cases on average she is right. As the hearer is wrong, both players receive 0 as the basic payoff. The speaker also receives the penalty of $-k$ because she is using a more complex linguistic system, so she is slightly worse off than the hearer.

Our next step is to generalize over how the game might play deriving the *expected payoffs* for each pairing of strategies in a random round. The idea is that if the basic game is played a very large number of times between two players, the average payoff approaches the expected payoff. We assume that each agent plays the role of speaker and hearer with equal likelihood.

Again for concreteness, let's compute the expected payoff for agents using *pcd* and *cd* strategies, who we just met playing one round of the basic game. First, consider cases when *pcd* is the speaker a_{pcd}, and *cd* is the hearer b_{cd}. Regardless of what signals a_{pcd} speaker sends, b_{cd} will always guess the nature state based only on the context C_{phen} or C_{struc}. This will lead to right guesses in 90% of cases. The average payoff for hearer b_{cd} in this pairing is thus 0.9. For speaker a_{pcd}, it is $0.9 - k$, as a_{pcd} also bears the penalty for using a two-form system. Now, when b_{cd} is the speaker and a_{pcd} is the hearer, the message sent is always the same: b_{cd} does not know how to send Y. In C_{phen}, hearer a_{pcd} always guesses **phen**, and thus is right in 90% of cases. In C_{struc}, a_{pcd} would have guessed **phen** upon hearing Y_{prog}, but speaker b_{cd} never uses that form. Therefore a_{pcd} always guesses **struc**, being right 90% of the time. Therefore for speaker b_{cd}, the success rate is again 0.9. For a_{pcd}, it is the same, as there is no penalty k for hearers. As our agents are equally likely to be speaker and hearer, we weigh their speaker and hearer expected payoffs by 50% to derive the overall expected payoff. For b_{cd}, it is 0.9. For a_{pcd}, it is $0.9 - k/2$. Note that it is better to be a b_{cd} than a a_{pcd} in this pair.

Once we do such computations for all pairings, we obtain the table of expected payoffs in (6). We will reinterpret this table as the table of actual payoffs of the Normal-Form Imperfective Game (NF ImpGame). We do this so that for the evolutionary analysis we could abstract away from the details of the basic game. The following assumptions are made to make this possible: we take each agent to interact with other agents a very high number of times; then we can take the expected payoff

punishment for the hearers, it simply changes the effective expected punishment for *pcd* and *em* agents by factor of 2.

of the basic game to be the actual payoff of our NF ImpGame built on top of it. In the basic game, payoffs are given out in individual rounds of communication; in the NF ImpGame, payoffs are given for overall communication between two agents.

(6) **Deo's payoff matrix A, or the Normal-Form Imperfective Game**

Strategies	*cd*	*pcd*	*em*	*cd′*
cd	0.9	0.9	0.7	0.9
pcd	$0.9 - \frac{k}{2}$	$0.95 - \frac{k}{2}$	$0.75 - \frac{k}{2}$	$0.7 - \frac{k}{2}$
em	$0.7 - \frac{k}{2}$	$0.75 - \frac{k}{2}$	$1 - \frac{k}{2}$	$0.7 - \frac{k}{2}$
cd′	0.9	0.7	0.7	0.9

3.2. **The Evolutionary Imperfective Game.** Our ultimate goal is to derive predictions about the innovation and propagation of different strategies in a population of agents. For that, [Deo, 2015] builds the Evolutionary (Ev) Imperfective Game (ImpGame) from the NF ImpGame in (6) coupled with a mutation matrix Q. The payoffs of the NF ImpGame are supposed to capture the *functional pressures* on language users to maintain or abandon a given grammar. In addition to the workings of functional pressures, *mutations* in Deo's model automatically shift some amount of agents with strategy i-parents into some other strategy.[7] That is, a Q_{ij} share of agents with an i-parent will end up acquiring grammar j instead of i, regardless of the relative communicative merits of i and j.

[Deo, 2015]'s mutation matrix Q, given in (7), captures Deo's expert judgement about how strong such misacquisition should be for different strategies. (In Section 4.1, we will discuss how the specific values in the matrix affect the evolutionary process in Deo's model.)

(7) **Deo's mutation matrix Q for the Evolutionary Imperfective Game**

Strategies	*cd*	*pcd*	*em*	*cd′*
cd	0.94	0.06	0	0
pcd	0.02	0.91	0.07	0
em	0	0	0.97	0.03
cd′	0	0	0	1

[7][Deo, 2015] describes mutation rates as "the barriers to the learnability of a strategy: cognitive or acquisition-related biases that might prevent its faithful transmission from parents to offspring" [Deo, 2015, p. 30]. It is easy to misunderstand the first part of this description as suggesting that mutation rates restrict the workings of functional pressures, serving as barriers. In fact, mutation rates are barriers to *faithful* transmission. In other words, they guarantee that regardless of functional pressures, some proportion of children agents in the model will *always* misacquire their parent's grammar.

In Deo's original setup, our population of speakers-hearers is assumed to be infinite. (We will abandon this assumption in Section 5, which will actually lead to much nicer predictions.) Time is discrete, and the population at each moment is taken to be a generation of language speakers. The population is described by vector $\bar{x} = \{x_i, x_j, ...\}$ with $i, j, ...$ being the available grammars, and x_i being the current share of i-speakers. The shares x_i always sum to 1, i.e. $\sum_i x_i = 1$.

Each speaker in the next generation learns their grammar from their single parent, or rather "linguistic parent". (Again, it is possible to relax this assumption of single parenthood. We will review that possibility in Section 6.3.) Deo identifies the parents and offspring with actual people; she assumes, together with much of the generative-grammar literature (cf. [Lightfoot, 1991], a.o.), that grammar may change only when children acquire a different grammar than their parents spoke. There is by now much evidence that this view is too restrictive, and that *some* grammar change continues to occur in adulthood, cf. [Raumolin-Brunberg, 2009], a.o., though language acquisition in the early childhood is still a different process from grammar change in adults, with the latter being apparently more limited in its scope. But if one assumes that speakers cannot acquire innovations in the progressive-imperfective system once they are adults, Deo's interpretation is indeed the only possible one. We will address it in more detail, and also suggest an alternative for those who are willing to give up on grammars changing only in the childhood, in Section 6.

Speaker-hearers within the population are randomly matched with each other to play the NF Imperfective Game. The expected payoff $f_i(\bar{x})$ for (an agent with) strategy i is thus the sum of the NF ImpGame payoffs A_{ij} from (6) for playing with strategy j multiplied by the share x_j of j-speakers in the population. The idea here is that in random matching, the chance to encounter a j-speaker is precisely their current population share x_j. So we weigh the payoffs A_{ij} from (6) by the shares x_j of each strategy, getting $f_i(\bar{x}) = \sum_j A_{ij} x_j$. These expected payoffs are interpreted as the current *fitnesses* of different grammars in the current population. Alternatively, one can view the same fitnesses as the average actual payoffs from every speaker-hearer playing the NF Imperfective Game with every other speaker-hearer. (Random matching is not the only possibility. For example, the agents in our population may live on a geographical map, or be organized into a social network. Then agents close to each other on the map or within the network could communicate more frequently. In this paper, we do not discuss such modifications, staying within the random matching case.)

The term *fitness* comes from biology. In biological evolutionary analyses, fitness may be defined simply as the (expected) number of offspring. E.g., an individual with fitness 1.2 will produce on average 1.2 times more offspring than an individual with fitness 1. When we transfer this notion into the study of language change, the literal number-of-offspring interpretation is not intuitively appealing. It is not plausible to think that a person's grammar for the progressive and imperfective would determine directly how many children they will have. For one thing, even if grammar may in principle affect the number of offspring (this is not *a priori* crazy as communicative efficiency could be one of the factors in reproductive success), we would not expect a single piece of the grammatical system to have a lot of effect on the number of children. [Deo, 2015] therefore assumes a cultural-evolution interpretation of fitness, where copying grammars is not equated with having biological offspring. Fitness will thus represent the number of "linguistic offspring": children who adopt someone's grammar through social interactions. But on both biological and cultural understanding, if we are not careful, our A in (6) may end up implying that when a community speaks a language with the categorical distinction between progressive and imperfective, they have up to 10% more children than a community whose grammar has a single imperfective form.[8] So even in the cultural-evolution setting, we still need to be careful about our interpretation of fitness. We postpone further discussion of the issue until Section 6.

So we have our fitness matrix A and mutation matrix Q. There are many ways to use those two in order to deterministically define an evolutionary trajectory given the starting state. Those different ways are called (deterministic) *evolutionary dynamics*. Deo chooses the **discrete-time replicator-mutator dynamics**, defined in (8), for her version of the Evolutionary Imperfective Game. Any evolutionary dynamics is built from certain assumptions about how the shares of different strategies (=grammars) change in time given A, Q and the current population state. We will discuss the linguistic implications of such assumptions at more length in Section 6, and here concentrate on introducing the dynamics itself.

(8) **The replicator-mutator discrete-time dynamics**

$$x'_i = \sum_j x_j \frac{f_j(\bar{x})}{\phi(\bar{x})} Q_{ji}$$

We can think of the replicator-mutator dynamics as working in two steps. At the first step, we determine how many offspring in the next generation will have a parent with grammar j. The dynamics says that this share is x_j (the current number of j-speakers in $\bar{x}$) times the ratio of j's fitness $f_j(\bar{x})$ to the average fitness of the population $\phi(\bar{x})$. The average fitness $\phi(\bar{x})$ is simply $\sum_j x_j f_j(\bar{x})$, representing the average

[8] Whether this is actually said depends on the assumptions about population size. To avoid the assumption of differential number of offspring, we need to fix the population size through external constraints, independent of fitnesses. Then fitnesses will become relative rather than absolute predictors of the number of offspring.

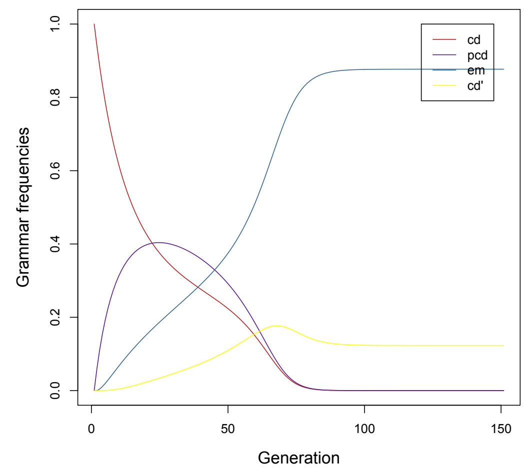
communicative success in the population.[9] Each j-individual produces $f_j(\bar{x})/\phi(\bar{x})$ children on average. Since our population is infinite, that average number will be the actual number as well. If strategy j's fitness $f_j(\bar{x})$ in the current population state $\bar{x}$ is greater than this $\phi(\bar{x})$, then there will be more children with j-parents than the share of such parents x_j. Conversely, if $f_j(\bar{x}) < \phi(\bar{x})$, the share of children with j-parents will be smaller than x_j.

At the second step, we determine which strategy each child ultimately adopts, using the mutation matrix Q. For the child of a j-parent, the j row of Q determines the probabilities with which it adopts one of the available strategies (=grammars). For concreteness, imagine that the share of children with an *em*-parent is 0.5 (recall that the overall population size is taken to always be 1 when we are dealing with an infinite population.) Then according to Q in (7), this will give rise to a share of 0.485 *em* children and 0.015 *cd′* children. Of course, the other half of the population could also produce *em* and *cd′* children. For instance, if the other 0.5 of the children have *cd′* parents, Deo's Q gives them no choice but to become *cd′* speakers, and the next generation will have 0.485 *em* speakers and and 0.515 *cd′* speakers. Note that what happens at the mutation step is independent of the communicative success of different strategies: even if grammar i is terrible compared to i, but there is a positive mutation rate from j to i, then some children of j-parents will end up speaking i. In biology, this is a natural setup: mutations in DNA happen by pure chance, and deleterious mutations are actually way more frequent than beneficial ones. In Deo's linguistic use of the mutation part, she assumes that mutation rates depend represent the actual probability of misacquiring a given grammar. Because of the general setup, it follows that a child is equally likely to acquire by mistake an innovative grammar when they are the first such innovator as when there are already plenty of speakers around with the innovative grammar.

Applying the formula of the RM dynamics from (8) to the starting population where all speakers use the first strategy, *cd*, we get the result in (9):

[9]How can a *sum* of fitness terms be the *average*? Note that x_j in the formula is the share of j-speakers in the population — that is, a number between 0 and 1. Therefore each term in the sum $\sum_j x_j f_j(\bar{x})$ would be less or equal to $f_j(\bar{x})$. In case the shares of each strategy are equal, the average fitness will be the arithmetic average of individual strategies' fitnesses. But when the shares are not equal, they serve as weights: if there are more i-speakers in $\bar{x}$ than j-speakers, then $f_i(\bar{x})$ affects $\phi(\bar{x})$ greater than $f_j(\bar{x})$.

(9) **Replicator-mutator dynamics with Deo's A and Q, and $k = 0.01$**




```
plot_imperfective_share(run_n_generations(c(1,0,0,0),
    function(x) replicator_mutator_discrete(x,A,Q), 200) )
```

Appendix A explains the structure of this command.

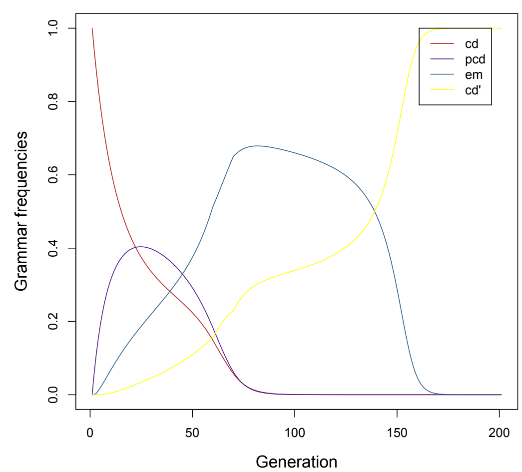
What are the notable features of the diagram in (9)? The first segment shows *cd* immediately losing ground to *pcd*, so the beginning of the first phase of the cycle is reproduced. However, *pcd* in this simulation never reaches above 41% of the population, and the share of the *em* grammar, with the categorical distribution of progressive and imperfective forms, starts to rise almost at the very beginning. Also, we see the share of the speakers with the last grammar, which only uses the Y_{prog} form regardless of the context, rising almost from the very beginning of the cycle. In the 46th generation, the share of the third strategy *em* overtakes that of *pcd*. The end point of the trajectory is an equilibrium where about 87.7% of the agents use the *em* grammar, while 12.3% speak the last-stage single-form grammar cd'.

This result is not terrible for the first try — after all, the grammars do succeed each other until a certain point where we get into an *em*-majority equilibrium. But there are many reasons to be dissatisfied. [Deo, 2015] targets one of those, namely the fact that the cycle does not end in an all-cd' population. She achieves a better result by applying a custom modification to the dynamics. Deo argues that once the share of *em*-speakers in the population is high, it is reasonable to expect out-mutationa to cd' to increase. The reason for that would be that the total frequency of Y_{prog} forms in the society would increase, and since it would be reasonable that the learners are biased to single-form systems, the mutations to the single-form cd'

should increase slightly in such a situation. Deo therefore changes the mutation rate $Q_{em,cd'}$ from 0.03 to 0.04 when x_{em} reaches the 0.5 share, and then to 0.05 when it reaches the 0.65 mark. She does not put the rate down as the *em* share decreases, which makes perfect sense under her reasoning: at this point, *em* speakers would be only turning to cd' speakers, and the overall usage of Y in the population will only increase even as the *em* share will decrease.

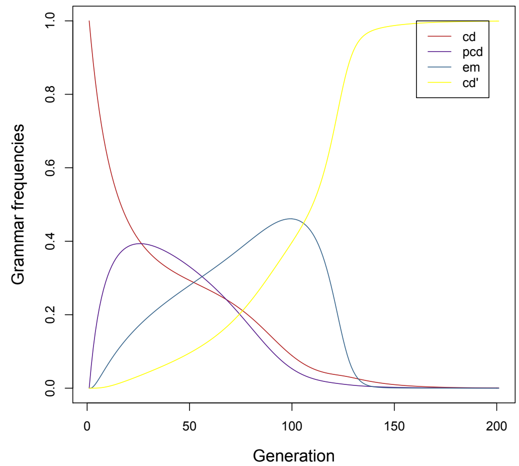
The trick works, as we can see in (10) (repeated from (2)): with such bonuses, cd' wins over *em* in the end. Adding such a bonus is an *ad hoc* move, however: the cost k factored into the payoff matrix A was already supposed to capture the additional attractiveness of single-form systems. In fact, as we will see in Section 4.3, changing the cost parameter k is enough to derive the takeover by grammar cd'. The diagrams in (9) and (10) are generated with $k = 0.01$, which seems to be the value [Deo, 2015] used in her diagrams. But with $k = 0.03$, the basic replicator-mutator dynamics without any additional bonuses produces the trajectory in (11), which is just as good as Deo's (10), but does not require any modifications of the general dynamics. In the next section, we will discuss in detail how the cost k affects the evolutionary trajectories. For now, it is enough to note that introducing hardwired bonuses for a certain strategy is not necessary to get the eventual dominance of cd'.

(10) **Replicator-mutator dynamics with Deo's A and Q, $k = 0.01$, and Deo's bonus for mutations $em \to cd'$:**




```
plot_imperfective_share(
    run_n_generations_RM_deo_bonus(x_start, 200, k=0.01) )
```

(11) **Replicator-mutator dynamics with Deo's A and Q, and $k = 0.03$:**




```
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,generate_A(k=0.03),Q),150) )
```

The RM dynamics chosen by [Deo, 2015] is one of the best known evolutionary dynamics, but it is not the only possible one. The replicator component of RM says that the share x_j grows proportionally to the ratio $f(\bar{x})/\phi(\bar{x})$. But the same basic intuition that a strategy with better payoffs should expand may be captured differently. For example, another well-known dynamics, given in (12), is the Brown-von Neumann-Nash (BNN) discrete-time dynamics, introduced in [Nash, 1951]. In BNN, the share of strategy j changes based on the absolute difference between j's fitness $f_j(\bar{x})$ and the average fitness $\phi(\bar{x})$.[10]

(12) **The Brown-von Neumann-Nash (BNN) discrete-time dynamics**

$$x'_i = \frac{x_i + \alpha[f_i(\bar{x}) - \phi(\bar{x})]_+}{1 + \sum_j \alpha[f_j(\bar{x}) - \phi(\bar{x})]_+}$$

$[y]_+$ is the sugar notation for $\mathbf{max}(0, y)$: that is, for positive y, $[y]_+$ is just y, and for non-positive y, $[y]_+$ is 0.

[10] We can think of obtaining the new share x'_i under BNN as a two-step process. First, we add to the old share x_i a new amount $\alpha[f_i(\bar{x}) - \phi(\bar{x})]_+ = \alpha[\hat{f}_i(\bar{x})]_+$, proportional to the fitness bonus that can be obtained by switching to i. We thus obtain "pseudo-shares" $x_i + \alpha[\hat{f}_i(\bar{x})]_+$ that form the numerator of the right-side of (12). Second, since we added exactly $\alpha[\hat{f}_j(\bar{x})]_+$ for each strategy j, our pseudo-shares sum up to $1 + \sum_j \alpha[\hat{f}_j(\bar{x})]_+$. We get true new shares x'_i by dividing the pseudo-shares by that amount, so that they are normalized to sum up exactly to 1. As the result, the shares of more successful strategies will increase, while the rest will be crowded out by their more successful competitors. Parameter α regulates how fast the more successful strategies proliferate.

It is not hard to add a mutation part to BNN, obtaining a BNN-mutator dynamics in (13). Just as in RM, in BNN-mutator a Q_{ki} share of children with k-parents will switch to grammar i, regardless of the relative functional merits of k and i. We thus obtain (13):

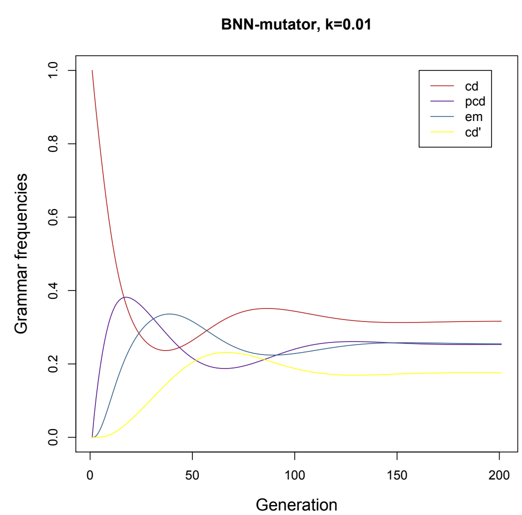
(13) **Discrete-time BNN-mutator dynamics**

$$x_i' = \sum_k \frac{x_k + \alpha[f_k(\bar{x}) - \phi(\bar{x})]_+}{1 + \sum_j \alpha[f_j(\bar{x}) - \phi(\bar{x})]_+} Q_{ki}$$

As they stand, neither RM or BNN-mutator dynamics is an obviously better choice for modeling the linguistic progressive-imperfective cycle. It is reasonable to assume that a strategy with better payoffs would grow, other things being equal. But how do we know in advance what the law for that growth should be? RM and BNN-mutator are just two possibilities out of infinitely many: it should be easy to conceive of many other ways in which the growth or decay of x_i could be dependent on its fitness and the average population fitness.

But the problem for the Evolutionary Imperfective Game is that those different dynamics lead to very different predictions. If the RM dynamics with Deo's A, Q, and cost $k = 0.01$ produces the trajectory in (9), the BNN-mutator dynamics with the same parameters produces the following trajectory:

(14) **BNN-mutator dynamics with Deo's A and Q, and $k = 0.01$:**




```
plot_imperfective_share(run_n_generations(x_start,
    function(x) BNN_mutator_discrete(x,A,Q),200) )
```

Thus under BNN-mutator, while the share of *pcd*-speakers initially increases, the system reaches an equilibrium with 31.6% of *cd*, 25.4% of *pcd*, 24.5% of *em*, and 17.5% of *cd′* speakers. The first grammar of the cycle continues to dominate, though it co-exists with all of the three other grammars. This bears no resemblance to what we would like to derive for the progressive-imperfective cycle.

Thus what may seem like an innocent choice between two sensible alternatives can lead to very different predictions. Therefore the choice of an evolutionary dynamics had better be grounded in substantial linguistic considerations. In the following Sections 4 and 5, we will only work with the replicator-mutator dynamics. We will return to the choice between different dynamics and to their linguistic interpretations in Section 6. In particular, we will explain in Section 6.5 why BNN-mutator just cannot work well in Deo's Ev ImpGame.

4. What fitness and mutation do in the Evolutionary Imperfective Game

The goal of this section is to show how the payoff matrix A and mutation matrix Q define particular equilibria of the evolutionary process in the Evolutionary Imperfective Game, assuming the replicator-mutator dynamics. To make this easier, we start by examining what happens if only payoffs (that is, fitnesses) in A or only mutations in Q affect the process. We will then show what happens when both A and Q contribute to the result, depending on their relative strength. But across all the settings we will consider, we will not get the desired modeling result: the clear sequence $cd \rightarrow pcd \rightarrow em \rightarrow cd'$. Either the system will go through to the final stage where all population uses a single form cd', but never pass through close-to-100% *pcd* and *em* stages, or else get stuck in a local equilibrium never getting to the all-*cd′* stage.

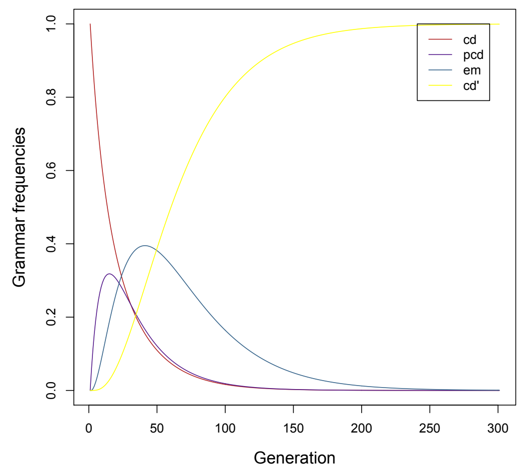
Thus on the one hand, we will see that changing the parameter values we can obtain predictions substantially different from Deo's, given in (10) above. Deo's trajectories are not robust to parameter changes. On the other hand, though we can obtain a wide range of various evolutionary trajectories, none of them achieves both modeling desiderata, namely (i) the completion of the cycle and (ii) the establishment of the "middle" grammars. Then in the next section, we will see how we can do better on those by dropping the assumption that our population is infinite.

4.1. What mutation does when left alone.

Recall that in the replicator-mutator dynamics, there are two "steps" that determine the shares of each grammar in the next generation: in the replicator, or fitness, step, the share of strategies performing better than the average $\phi(\bar{x})$ increases, and then in the mutator step, all children of i-parents determine their acquired strategy according to the i row of mutation matrix Q.

What would happen if we omit the fitness step? In other words, if we only have mutations, but not functional pressures, what would the Evolutionary Imperfective Game do? The plot in (15) shows that:[11]

(15) **The evolutionary trajectory derived by mutations in Deo's Q, without any functional considerations:**




```
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,A_all_equal,Q),300) )
```

The diagram we obtained is conspicuously close to Deo's original predictions in (10)! (15) shows that mutation alone is actually enough to push the starting 100%-*cd* population to the final stage of the cycle. Moreover, the role of fitness in Deo's original analysis is actually quite small.

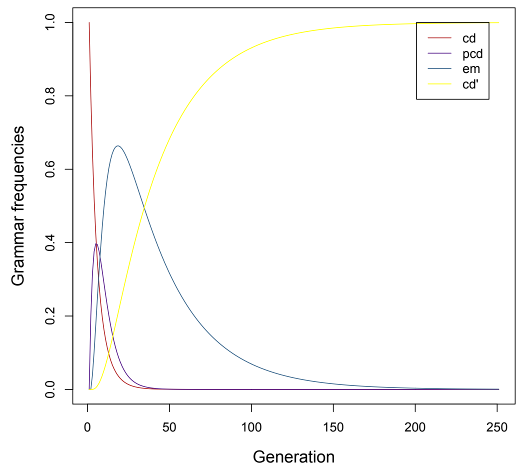
Though the picture in (15) is slightly different from Deo's original predictions in (10), those differences may be made smaller by changing Q. For example, the share of *em* speakers in Deo's (10) reaches above 0.65. If we slightly change the mutation rates out of strategies *cd* and *pcd*, creating $Q1$, we can achieve that as well:

[11]We generate (15) using the function for the RM dynamics, but with the payoff matrix with one and the same value in all cells (we used 1, but the specific number does not matter.) We could have defined another, simpler dynamics function just for computing mutation-only evolution; but it is easy to check that it would be equivalent to using either RM or BNN-mutator with all payoffs being equal.

(16) **Mutation-only evolutionary trajectory with $Q1$,**

where $Q1$ =

Strategies	cd	pcd	em	cd'
cd	0.8	0.2	0	0
pcd	0.02	0.78	0.2	0
em	0	0	0.97	0.03
cd'	0	0	0	1




```
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x_start,A_all_equal,Q1),250) )
```

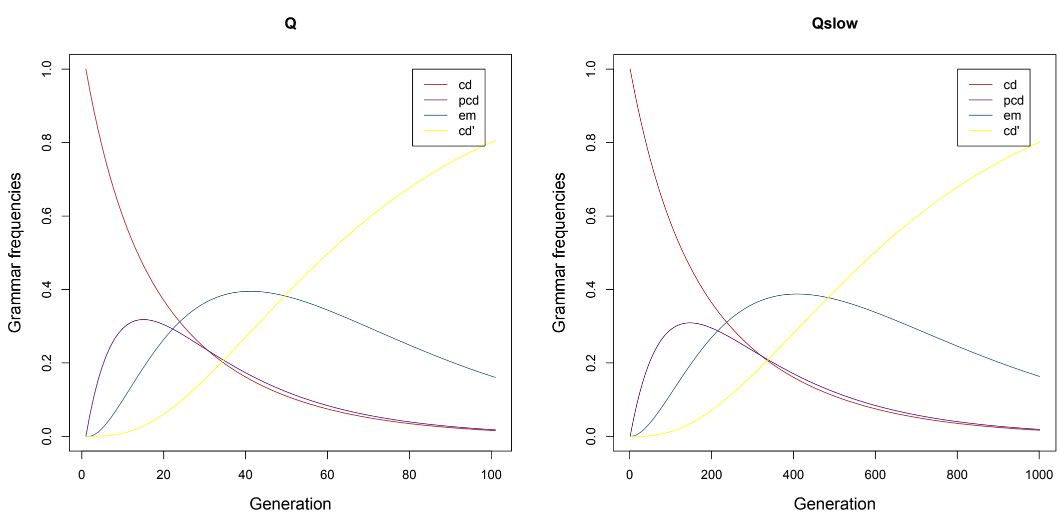
When we discussed in Section 1 what evolutionary trajectories for the progressive-imperfective circle we may want to derive given the current understanding of the data in the literature, we noted two crucial desiderata: (i) reaching the final 100%-cd' stage, and (ii) predicting close-to-100% establishment of the intermediate grammars pcd and em. Deo's original analysis captures (i), but not (ii). Examining the behavior of mutation-only trajectories, we can see that mutation alone, as defined in Deo's Q, already derives the desideratum (i). In other words, the whole complex game-theoretic analysis in terms of the basic Imperfective Game and Evolutionary Imperfective Game that we repeated from [Deo, 2015] in Section 3 is not necessary to get (i). There is thus no predictive benefit from that more complex model as it stands. If the model could derive (ii) in addition to (i), that would be a different story: mutation alone cannot do that. In what follows, we will strive towards that goal. We will not be unequivocally successful, but will still fare better than it is possible with mutation alone, thus justifying the value of introducing a more complex model of Deo's type.

It is easy to see why the mutation-only evolutionary process defined by Q will converge on an all-cd' population. Imagine an infinite chain of speakers where each next speaker is the child of the current one. The grammar of agent $n+1$ is probabilistically dependent only on the grammar of their parent n and the probabilities that determine $(n+1)$'s grammar are given by Q. (In other words, we have a Markov chain with Q as its transition matrix.) The probability to get to *pcd* from *cd* is positive, and so are the probabilities to get from *pcd* to *em* and from *em* to cd'. Therefore sooner or later, our chain will definitely pass through the cd' state. But there are no out-mutations from cd'. Therefore given Q, every such infinite chain will eventually end up being locked in the cd' state.

The speed with which the convergence to cd' will happen depends on out-mutation rates. It is instructive to compare the evolutionary process under the original Q with the process under Q_{slow}, featuring out-mutation rates 10 times lower than in Q. The intuition from (17) will prove handy when later in Section 4 we will study the interplay of mutation and fitness varying the pace of mutation. Though the process with Q_{slow} is about 10 times slower, but other than that, it is very similar (though not completely identical) to the process we get using the original Q. Therefore it will be OK for us to substitute Q_{slow} for Q when it will be convenient: the general properties which the two impose on the evolutionary process are very similar.

(17) **Mutation-only evolutionary trajectory with Q vs. with Q_{slow},**

$$\text{where } Q_{slow} \;=\; \begin{array}{l|cccc} \text{Strategies} & cd & pcd & em & cd' \\ \hline cd & 0.994 & 0.006 & 0 & 0 \\ pcd & 0.002 & 0.991 & 0.07 & 0 \\ em & 0 & 0 & 0.997 & 0.003 \\ cd' & 0 & 0 & 0 & 1 \end{array}$$




```
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x_start,A_all_equal,Q),100) )
```

```
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x_start,A_all_equal,Q_slow),1000) )
```

4.2. **What fitness does when left alone.** We have just seen that mutation matrix Q, when left to its own devices, will sooner or later lead the population to the final stage. On the other hand, the fitness matrix A in Deo's NF ImpGame is such that it creates islands of stability for each strategy when its share in the population is close to 100%.

Consider the replicator dynamics, which is simply the replicator-mutator dynamics without the mutation component:

(18) **The replicator discrete-time dynamics:**

$$x'_i = x_i \frac{f_i(\bar{x})}{\phi(\bar{x})}$$

First, suppose that our population $\bar{x}$ fully consists of *pcd* speakers. Then for any other strategy i, $x_i = 0$, and this means that x'_i will also be 0. Therefore it is uninteresting to consider cases where any strategy has the 100% share under this dynamics. Let us instead consider $\bar{x}$ where $x_{pcd} = 0.97$, and all other strategies have shares of 0.01. Fitnesses of different strategies in this population will be as follows:

(19) For $x = (0.01, 0.97, 0.01, 0.01)$:

f_{cd}	f_{pcd}	f_{em}	$f_{cd'}$
0.898	0.94	0.7465	0.704

It is easy to see why the fitness of *pcd* is the highest: according to the fitness matrix A, every strategy is a best response to itself. Therefore in a population where some strategy's share is close to 100%, no other strategy can be at an advantage. As the evolutionary process under the replicator dynamics is guided by fitnesses alone, this means that every strategy creates a basin of attraction around the population state where all speakers use it.

At the same time, there also exist equilibria where several strategies co-exist. Let us find out if there is an equilibrium where strategies *cd* and *pcd* coexist. Since we are interested in those two, we can forget about *em* and *cd'*: if their shares are 0, they will stay at 0. (Note that this trick would not have been possible if we had the replicator-mutator dynamics, because there, a strategy that is currently absent may arise through mutation.) We can find the equilibrium by solving the system in (20). Its first equation is obtained from the definition of the replicator dynamics by saying that the current share x_{cd} also serves as the next share of that strategy.

(20) $x_{cd} = x_{cd} \frac{f_{cd}(x_{cd}, x_{pcd}, 0, 0)}{\phi(x_{cd}, x_{pcd}, 0, 0)}$
$x_{cd} + x_{pcd} = 1$

When $x_{cd} \neq 0$, this system amounts to the quadratic equation shown in (21) together with its two roots. And the third root of the original system in (20) above is $x_{cd} = 0$.

(21) $0.05x_{cd}^2 + (\frac{k}{2} - 0.1)x_{cd} + (0.05 - \frac{k}{2}) = 0$

When $k = 0.01$, $x_{cd,1} = 0.9$, $x_{cd,2} = 1$.

Indeed, we can check that when we apply the mutator dynamics `mutator_discrete` to $x = (0.9, 0.1, 0, 0)$ with the payoff matrix A, we get back the same x. As there are no other roots of the system in (21), there are only three equilibria for populations with only *cd* and *pcd* speakers under fitness-only replicator evolution: $(1, 0, 0, 0)$, $(0.9, 0.1, 0, 0)$, and $(0, 1, 0, 0)$.

In the similar manner, we can algebraically find the equilibria with more non-0-share strategies. This involves solving a system of linear equations which are obtained just as the one in (20). For Deo's A, all the equilibria in the replicator dynamics are given in (22).[12] Note that in some equilibria, any shares for strategies *cd* and *cd′* would do, as long as all the share in the population state sum up to 1. This happens when the payoff profiles of two strategies are exactly the same. For Deo's A, that is the case when *cd* and *cd′* play only against each other, or also against *em*.

(22) **Equilibria for Ev ImpGame under the replicator dynamics:**

- 1-grammar equilibria are trivial, and exist for all four strategies.
- 2-grammar equilibria:

x_{cd}	x_{pcd}	x_{em}	$x_{cd'}$
0.9	0.1	0	0
0	$\frac{5}{9}$	$\frac{4}{9}$	0
0	0	0.41	0.59
0.59	0	0.41	0
a	0	0	$1-a$
0	$\frac{41}{90}$	0	$\frac{49}{90}$

[12] The equilibria for the replicator dynamics may be found for an arbitrary (small) A with the functions `equilibria_for_N_strategies` in the source code file `solving_equilibria.r`. One can always double-check if a found solution is indeed an equilibrium by creating the corresponding population-state vector, and applying to it `replicator`$_d$`iscrete` with the same A.

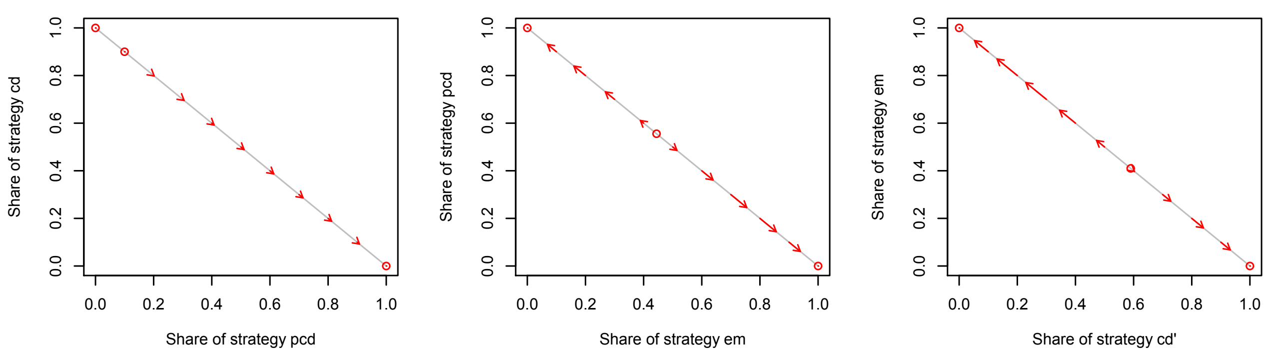
- 3-grammar equilibria:

x_{cd}	x_{pcd}	x_{em}	$x_{cd'}$
a	0	0.41	0.59 - a
0	≈0.315	≈0.252	≈0.432

- There are no 4-grammar equilibria.

In addition to identifying equilibria themselves, it is often useful to picture the direction into which the evolutionary process is set to go from different starting conditions. The diagrams in (23) provide such an illustration for the forces between the pairs of strategies $\langle cd, pcd\rangle$, $\langle pcd, em\rangle$, and $\langle em, cd'\rangle$ — in other words, for each pair where the second strategy is supposed to gradually replace the first one in the actual linguistic progressive-imperfective cycle. In each diagram, there is a line from point (1,0) to point (0,1), representing all possible two-strategy population states. We select several equidistant points on this line, and compute what each point will be transformed into in the next step of the replicator dynamics. Then we draw an arrow for each such transformation. Because the equilibria of the discrete-time replicator dynamics are determined by a system of linear equations, either there will be a single internal equilibrium, no such equilibria, or every point on the line would be an equilibrium. We mark the equilibria with circles.

(23) **Forces between $\langle cd, pcd\rangle$, $\langle pcd, em\rangle$, and $\langle em, cd'\rangle$ pairs of strategies in the replicator-dynamics (=fitness-only) Evolutionary Imperfective Game with $k = 0.01$:**

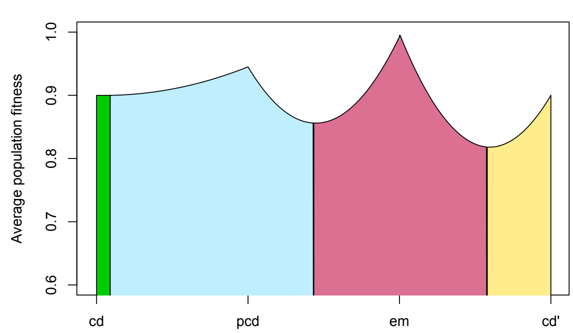



```
forces_btw_2_strategies_replicator(1,2,"cd","pcd",A)
forces_btw_2_strategies_replicator(2,3,"pcd","em",A)
forces_btw_2_strategies_replicator(3,4,"em","cd'",A)
```

The visualization in (23) suggests that the internal equilibria are unstable, which is generally true for the case when the two strategies fare better playing against themselves than against each other. The positions of those equilibria thus mark the borders of each strategy's basin of attraction, for the 2-strategy case. For example, we can see that once *cd* gets below 0.9, the population state will converge to 100% of *pcd*, so *cd* has quite a small basin. In contrast to that, the basin for *em* starts at 4/9 against *pcd*, and at 0.41 against cd'.

Another way to visualize fitness pressures is through the means of a **fitness landscape**, provided in (24). Each point on the x-axis of the diagram corresponds to a 2-strategy population state. E.g., the point in the middle of the section between the *cd* and *pcd* marks on the diagram corresponds to the population state (0.5, 0.5, 0, 0). The height on the y-axis is the average fitness ϕ of the current population. We see that the 1-strategy states constitute peaks in the landscape. This means that reaching 100% of some strategy, we get to the local optimum of fitness. The unstable internal equilibria lie in the valleys of the fitness landscape. On the two sides of such an equilibrium, fitness considerations push the populations to move to different peaks. (It is not guaranteed that the unstable equilibrium will be in the lowest point of the valley. So even though evolution in this setting is guided by fitness alone, the average fitness of a population can sometimes decrease.) The absolute height of the peak does not matter for the direction of such movement. The fitness-only evolution is "stupid": it would always go to the local peak even when there is in principle a higher one. If we want to get to that higher peak, some additional factor needs to enter into play — such as mutation, which pushes some part of the population into another strategy, or some form of stochasticity that creates fluctuations of the population shares, which would sometimes be enough to jump across the unstable equilibrium in the valley.

(24) **The (2-strategy) fitness landscape of the replicator-dynamics (=fitness-only) Evolutionary Imperfective Game with $k = 0.01$:**




```
fitness_landscape(A, c("cd", "pcd", "em", "cd’"), y_region=c(0.6, 1.0))
```

A 2-dimensional fitness landscape is in the general case only of limited use for a 4-strategy game: there may be non-trivial interactions between the four strategies that would be invisible in the 2-dimensional setup. But the payoff matrix A of the NF ImpGame, given above in (6), is such that every strategy will always be one of the best responses to itself. That implies that the average fitness of a mixed-strategy population cannot exceed the highest of the fitnesses of the strategies in it. In other words, there are no peaks in the four-dimensional fitness landscape that are higher than the ones we see in e.g. (24). This won't always be so, but in the Imperfective Game, it is.

We can thus gain useful insights from examining such 2-strategy fitness landscapes. For example, in (24) we can observe that the highest peak is at the strategy *em*. Furthermore, the basin of attraction for the initial strategy *cd*, marked in green, is very small. Later on, we will see how these properties affect the evolutionary dynamics even in the presence of mutation.

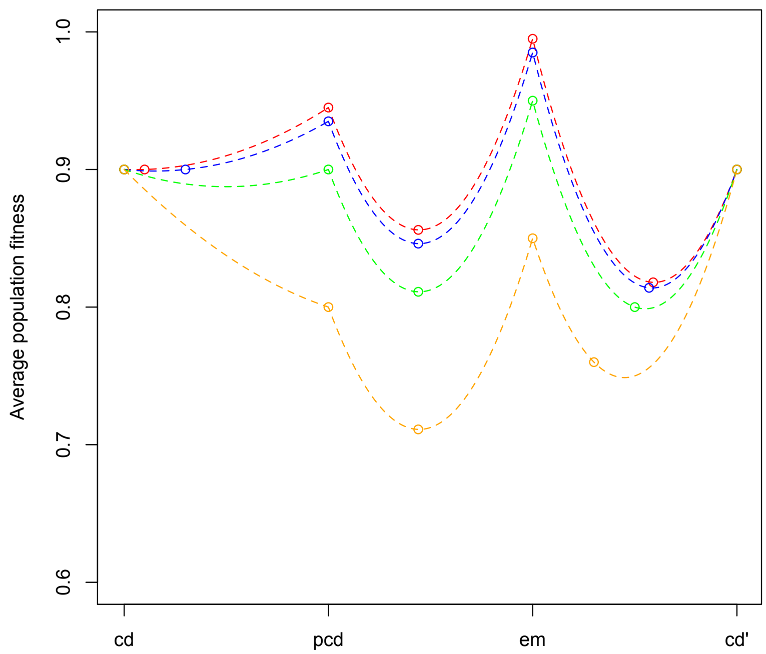
Fitness landscapes make it easy to see what effects the cost k of having a grammar with 2 forms has. In (25), we show four landscapes generates by NF ImpGame matrices with different k. Two facts should be noted. First, increasing k lowers the height of the internal peaks. Indeed, k punishes the strategies *pcd* and *em* for using two forms no matter how efficient they are otherwise. With a low k, the additional communicative efficiency brought by disambiguation makes the internal peaks higher than the two side peaks where the hearers can only guess the intentions of the speakers based on the extralinguistic context. But as cost k increases, it starts to outweigh the gain in communicative efficiency.

Second, while the internal equilibrium between *pcd* and *em* remains in place across the four values of k (because these two strategies bear cost k to the same extent), the equilibria between *cd* and *pcd*, and *em* and *cd′* shift in favor of *cd* and *cd′* respectively. For *pcd*, the internal equilibrium disappears completely with $k = 0.1$, which means that even a tiny amount of *cd* speakers mixed into a *pcd* population will be enough to cause a complete takeover by *cd*.[13] For *em*, the effect is not that radical, but the basin of attraction shrinks significantly: with $k = 0.01$, *em* will dominate as long as the share of *cd′* is lower than 59%, but with $k = 0.3$, *em* only wins when the share of *cd′* is lower than 30%.

[13]The reader who would like an illustration may wish to run the following command:
```
plot_imperfective_share(run_n_generations(c(0.01,0.99,0,0), function(x)
replicator_discrete(x,generate_A(0.3)), 100))
```

(25) **Comparing (2-strategy) fitness landscapes of the fitness-only Ev ImpGame with different k.**

Red: $k = 0.01$, blue: $k = 0.03$, green: $k = 0.1$, orange: $k = 0.3$.
Dots indicate equilibria.



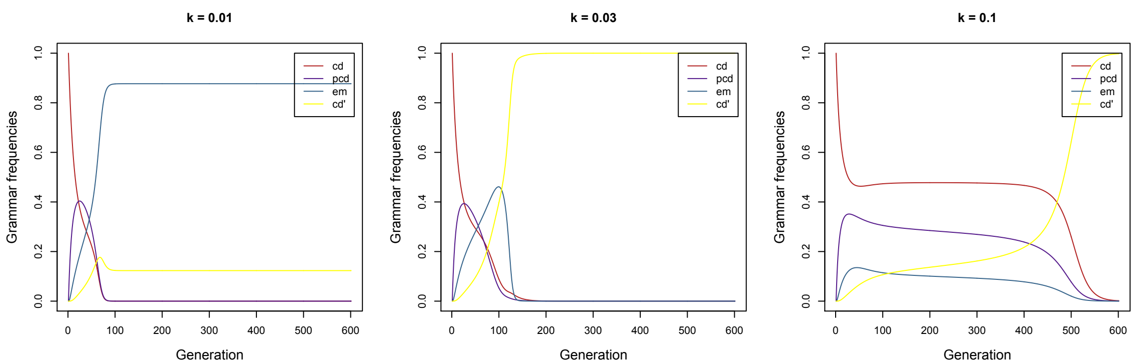

```
compare_fitness_landscapes(list(generate_A(0.01), generate_A(0.03),
    generate_A(0.1), generate_A(0.3)),
    c("cd", "pcd", "em", "cd'"), y_region = c(0.6, 1))
```

Note that in this section, we did not present explicitly any evolutionary trajectories. They are easy to generate using the function `replicator_discrete` that computes the fitness-only replicator dynamics. But examining such trajectories is not necessary as our analysis of the equilibria, forces between pairs of strategies, and fitness landscapes already tells us what we might want to know. For example, we have learned that with fitness only, we are never going to get from the 100% dominance of one strategy to another. Running explicit evolutionary simulations is not necessary to understand the evolutionary process in this case (though it can still be useful for building better intuitions).

4.3. **Combining fitness and mutation.** Let's sum up what we have just established. When mutation alone is at work, the Imperfective Game relatively quickly cycles through the four strategies, reaching the final *cd′*. But the middle strategies never get close to a complete takeover with 100% of speakers using them. When fitness works alone, the populations with 100% of the speakers of some strategy are stable by definition; but even more importantly, these 100% states (which can also be called *pure* states) often constitute peaks that serve as attractors for large neighborhoods of states. Thus just intuitively, we could think that combining the forces of mutation and selection *could* produce both desideratum (i), namely the convergence to the fourth grammar *cd′* of the cycle, and desideratum (ii), namely a nearly-100% establishment of the middle grammars *pcd* and *em*.

In reality, this does not really work this way. Consider first what happens in the Ev ImpGame when we plug into the replicator-mutator dynamics Deo's mutation matrix Q and her NF ImpGame's matrices under cost k of 0.01, 0.03, and 0.1:

(26) **Ev ImpGame under the replicator-mutator dynamics with Deo's Q and different costs k of 0.01, 0.03 and 0.1, determining the exact shape of Deo's A:**




```
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,A,Q), 600))
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,generate_A(0.03),Q), 600))
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,generate_A(0.1),Q), 600))
```

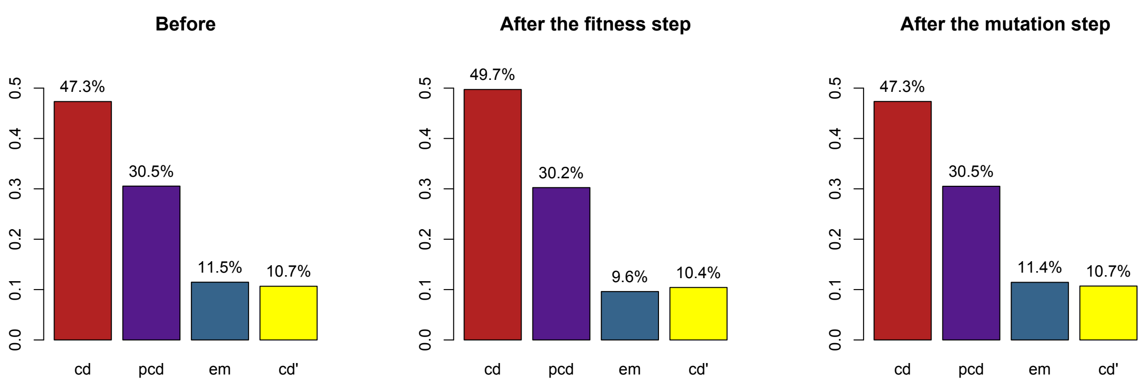
When k is small, the middle grammars *pcd* and *em* do better. In fact, in the leftmost diagram in (26) we see that *em* is doing so well that the process never goes towards the establishment of a full-*cd′* population. Instead, the equilibrium reached features more than 80% of *em* speakers coexisting with less than 20% of *cd′* speakers.

Increasing k to 0.03, we make life harder for the middle grammars, and the process quickly passes through the local population peaks of *pcd* and *em*, each below 50%, before converging to an all-*cd′* state. As we discussed above in Section 3.2, such settings for k allow us to reproduce the important features of Deo's original evolutionary trajectory without adding extra bonuses to cd' that depart from the general replicator-mutator equation.

If we increase k further to 0.1, the middle grammars become miserable. We can see in (25) that the fitness of *em* is still quite good in a population where *em*'s share approaches 100%, but starting from an all-*cd* population, we are never going to get there. Both *pcd* and *em* grammars in this setting never get a larger share of the population than the initial grammar *cd*. Mutations turn *cd* speakers into *pcd* speakers, and those into *em*, but because of their poor fitness against both *cd* and cd', those middle-grammar speakers leave very few offspring, and just serve as material for getting more and more cd' speakers through out-mutations.

To get a better grasp on the mechanics here, let's imagine the population state as a collection of four bins with liquid. The bins correspond to strategies, and the amount of liquid in each bin represents the corresponding population share. The replicator-mutator dynamics can be imagined as a two-step process. First, the amount of liquid in each bin i changes proportionally to $f_i(\bar{x})/\phi(\bar{x})$, that is the ratio of the strategy's fitness to the average fitness. Second, for each i and j we take the Q_{ij} share of the liquid in bin i and transfer it to bin j.

(27) **Fitness and mutation steps leading from the 100th to the 101st generations under Q and A generated by $k = 0.1$:**

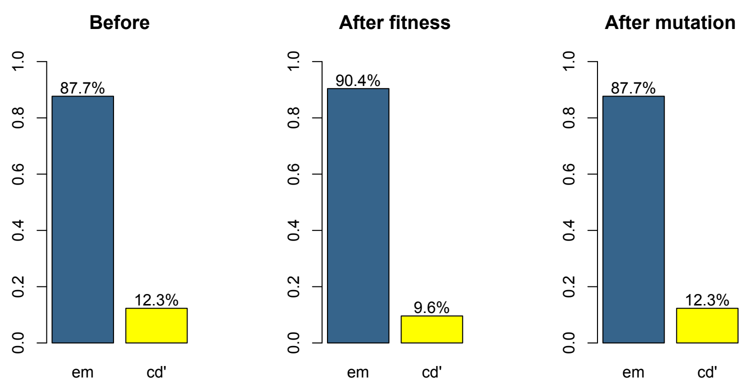


(27) shows how this works at the 100th generation after an all-*cd* start with $k = 0.1$ (i.e. in the process shown in the rightmost diagram in (26)). After the fitness step, the share of the original strategy *cd* increases, while all three others decrease. At this population state, only *cd* has fitness above the average (this is so because its share in the population is high, and it plays well against itself). But regardless of the fitnesses, at the mutation step we transfer agents between the bins according to Q. It so happens that the excess gained by *cd* vanishes after that, having been transferred into *pcd*. The other strategies also experience shifts between each other. After the

mutation step, everyone ends up almost where they were, but there are changes that are often too small to show in the graph; for example, the share of cd' actually increases slightly from 10.66% to 10.71%. Even though the fitness of cd'-agents is lower than the average, and thus their share declines accordingly at the fitness step, thanks to the in-mutation from em, their overall number actually increases slightly. In 500 more generations or so, this would lead to a complete takeover by cd'.

As another example, let us examine the equilibrium between pcd and cd' to which the process converges with Q and $k = 0.01$ (i.e. as shown in the leftmost diagram in (26)). We can see in (28) that at the fitness step cd' loses ground. It goes from 12.3% down to 9.6%, because em is in the majority, and cd' plays worse against em than em itself does. But then at the mutation step, no remaining cd' speakers are converted to em speakers, as Deo's matrix Q does not allow that. However, 3% of the em speakers are converted into cd' speakers. (More accurately, 3% of the em parents produce a cd' offspring.) This brings the system back to exactly where we were before the fitness step.

(28) **Fitness and mutation steps at the equilibrium to which Ev ImpGame converges with Q and A generated by $k = 0.01$:**



It should now become clearer what happens when we change k, as shown in (26). Consider the population state we started with in (28), which is an equilibrium under $k = 0.01$. When k is increased, em becomes worse off, and cd' in this state fares relatively better. With $k > 0.01$, cd' will decline less at the fitness step than we have in (28). But at the mutation step, 3% of the em speakers would still be transferred to bin cd'. Therefore this time, the out-mutated amount will be greater than the amount cd' lost due to fitness. The process will move towards a state with more cd' speakers. If k is increased significantly, this will lead to a complete takeover by cd'.

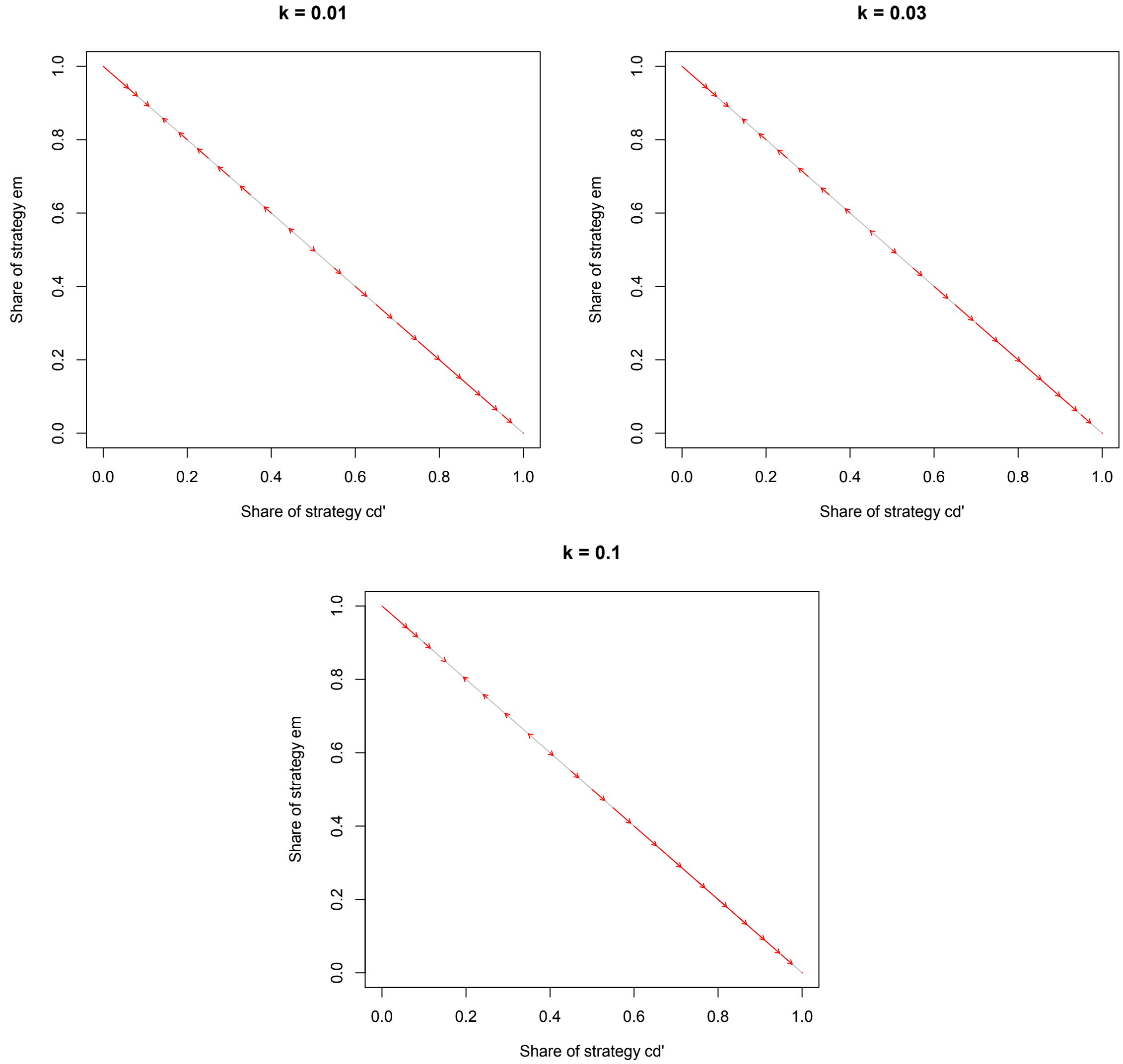
Finding the equilibria of the discrete-time replicator-mutator dynamics is a harder task than with the replicator dynamics. In the latter case, we only needed to solve a linear system. For RM, the system will be polynomial. The theory of solving systems of polynomial equations is well developed (after all, such systems arise frequently in many areas of science), but is much more mathematically complicated than the simple linear-algebraic methods for solving linear systems. To put it bluntly, R is just not the right tool for doing this kind of math (software like Maple or MATLAB should be used instead.)[14] For the purposes of this paper, we will not attempt to find equilibria directly — but because of the special properties of Deo's Q, we will be able to derive a partial answer indirectly.

The pair-of-strategies analysis of forces, which we have done above for the fitness-only replicator dynamics, is also of very limited value here. Consider again the evolutionary trajectories in (26). At most crucial moments, they involve an interaction between all the four strategies in the game. Therefore we cannot get direct insight into the evolutionary process by examining how two strategies play against each other. The only such analysis which is useful is between *em* and *cd′*: as there are no mutations out of those two into *cd* and *pcd* in Deo's Q, once we get to a state with only *em* and *cd′*, we can confine out attention to them. Since we can establish from the RM equations that the behavior of the system is monotonic enough, a forces analysis at different k is suggestive of where the internal equilibria between *em* and *cd′* are, and how a system with just those two would develop:

[14]To check this, I tested how the numerical-approximation tools in R package BB would fare. For Deo's Q and A with $k = 0.01$, they simply fail to find other equilibria than (0, 0, 0, 1), with 100% of *cd′* speakers. The internal equilibrium with $x_{em} \approx 87.7\%$ and $x_{cd'} \approx 12.3\%$ cf. the leftmost diagram in (26), and (28)) is not found even from the starting point very close to it. To check this oneself, one might run the following commands (the first of them installs the BB package):

```
install.packages("BB")
source("YOURPATH/ImpGame_solving_equilibria.r")
equilibria_for_RM(A,Q)
```

(29) **Forces between *em* and *cd′* with *Q* and *k* of 0.01, 0.03 and 0.1:**




```
forces_btw_2_strategies_generic(3,4,"em","cd’",
    function(x) replicator_mutator_discrete(x,A,Q),4, step=0.05)
forces_btw_2_strategies_generic(3,4,"em","cd’",
    function(x) replicator_mutator_discrete(x,A,Q),4, step=0.05)
forces_btw_2_strategies_generic(3,4,"em","cd’",
    function(x) replicator_mutator_discrete(x,A,Q),4, step=0.05)
```

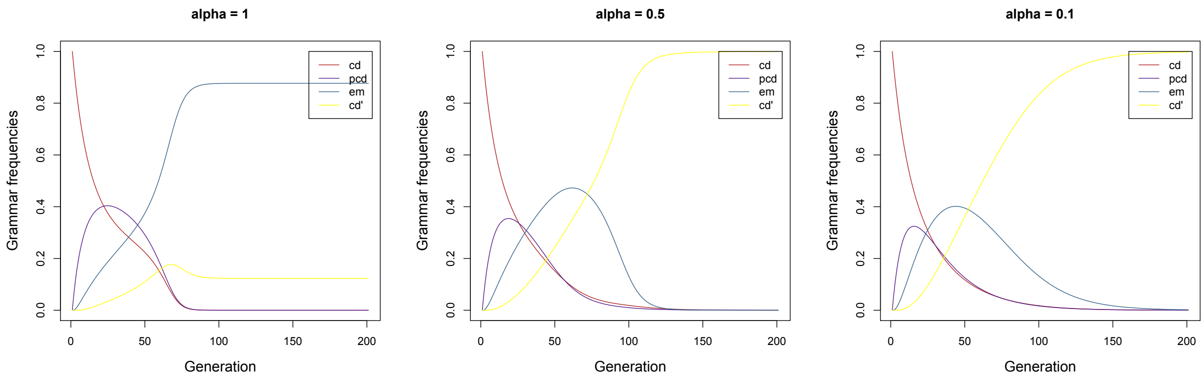
Under all three settings of k in (29), there are three distinct regions. First, a population with a high number of *em* speakers always moves downward towards an internal equilibrium. This equilibrium is stable: its immediate neighborhood converges to it from both sides. It is to such an equilibrium that the evolutionary process converges in the leftmost diagram of (26). The position of this equilibrium clearly varies somewhat, having more *cd′* speakers at higher k — just as we can expect, given that higher values of k punish the two-form grammar *em*. Second, the diagrams suggest that there is another internal equilibrium in all three cases. On the right from it, the basin of attraction towards a fully-*cd′* population starts. Again, its position shifts as we can expect: *cd′*'s basin of attraction is wider under higher k.

This analysis allows us to see the first important modeling problem we encounter. Recall that we want to derive both (i) the final convergence to *cd′*, and (ii) the nearly-full establishment of two middle strategies, *pcd* and *em*, at the earlier stages of the cycle. Forget about the establishment of *pcd* for the moment, and consider *em*. The diagrams in (29) suggest that we are in serious trouble here. If we get close to 100% of *em* speakers at some point, this means that the system will never reach the 100%-*cd′* state. It will be stuck instead in the internal stable equilibrium where *em* and *cd′* happily coexist. If, on the other hand, we derive the final takeover by *cd′*, this means that the share of *em* could never have been greater than about 50%: otherwise we would not have gotten into *cd′*'s basin of attraction. In other words, our desiderata (i) and (ii) appear just incompatible within this model and such parameter values!

Can we change some parameters of the model to derive a different behavior? To some extent, yes, but unfortunately, that will not get for us (i) and (ii) together. Fitness considerations push the evolutionary process towards pure states, while mutation works to move the population gradually into the final *cd′* state. The resulting trajectory depends on the balance between these two forces. We can vary the force of selection through an additional parameter α that scales up the fitness bonus to growth that every strategy receives.[15] If we decrease α, the trajectory becomes closer to the mutation-only trajectory:

[15] We define α as the regulator of how far from 1 a given fitness bonus can go, where 1 corresponds to the 1-for-1 reproduction. The α-regulated versions of RM-based dynamics are obtained by replacing $\frac{f_i(\bar{x})}{\phi(\bar{x})}$ with $1-\alpha(1-\frac{f_i(\bar{x})}{\phi(\bar{x})})$. It is easy to check that with $\alpha=1$, this is simply $\frac{f_i(\bar{x})}{\phi(\bar{x})}$. As an example, suppose $\frac{f_i(\bar{x})}{\phi(\bar{x})}=1.1$ and $\frac{f_j(\bar{x})}{\phi(\bar{x})}=0.9$, and take $\alpha=2$. Then the adjusted fitness bonus will be 1.2 for i (instead of 1.1) and 0.8 for j (instead of 0.9).

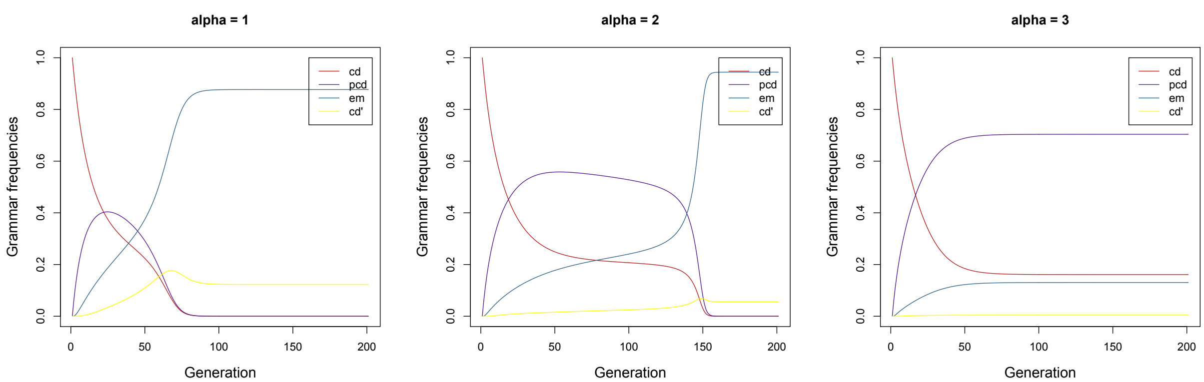
(30) **With lower α, we get closer to mutation-only evolution (Deo's Q, A with $k = 0.01$):**




```
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,A,Q,alpha=1), 200))
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,A,Q,alpha=0.5), 200))
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,A,Q,alpha=0.1), 200))
```

On the other hand, if we tip the balance more towards fitness, this may create internal equilibria characterized by the dominance of some strategy. In such an equilibrium, the outflow from the dominant strategy because of mutation is exactly compensated by its better growth. This is illustrated for *pcd* in the rightmost diagram in (31):

(31) **With larger α, fitness slows down the transfer of agents due to mutation (Deo's Q, A with $k = 0.01$):**

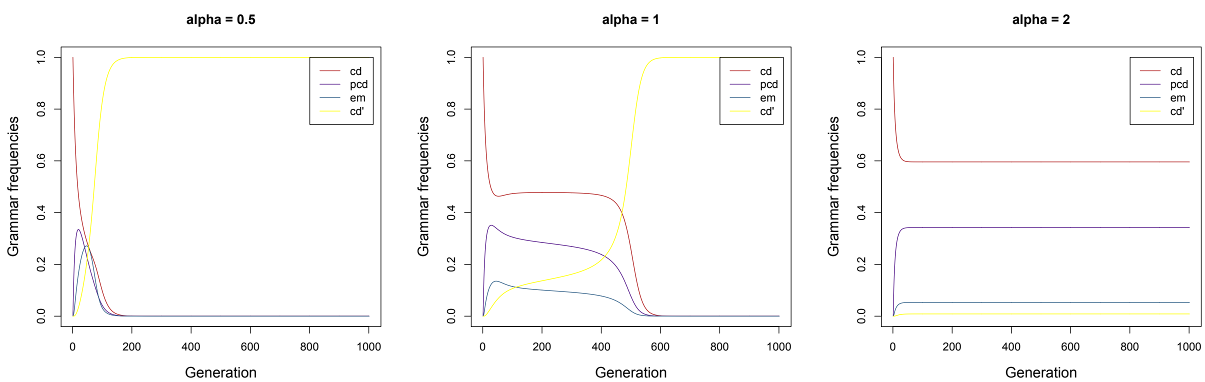



```
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,A,Q,alpha=1), 200))
plot_imperfective_share(run_n_generations(x_start, function(x)
```

```
    replicator_mutator_discrete(x,A,Q,alpha=2), 200))
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,A,Q,alpha=3), 200))
```

The exact effect of varying α will, of course, depend on the value of k, as k also affects both the equilibria's positions and the forces applying to different states. In (31) above, we use $k = 0.01$, which makes the fitness peaks of the middle strategies higher than those of the outer strategies (recall our fitness landscapes). With $k = 0.1$, the middle peaks become lower, and we get somewhat different trajectories. But the main result remains: either we get stuck in an internal equilibrium where one strategy is dominant, but the others are also present; or we arrive at a 100%-*cd′* population. What changes is only the position of the equilibria and the specific trajectories towards them.

(32) **Under $k = 0.1$, varying α has a similar effect, though the exact equilibria differ ($\alpha = 0.5; 1; 2$):**

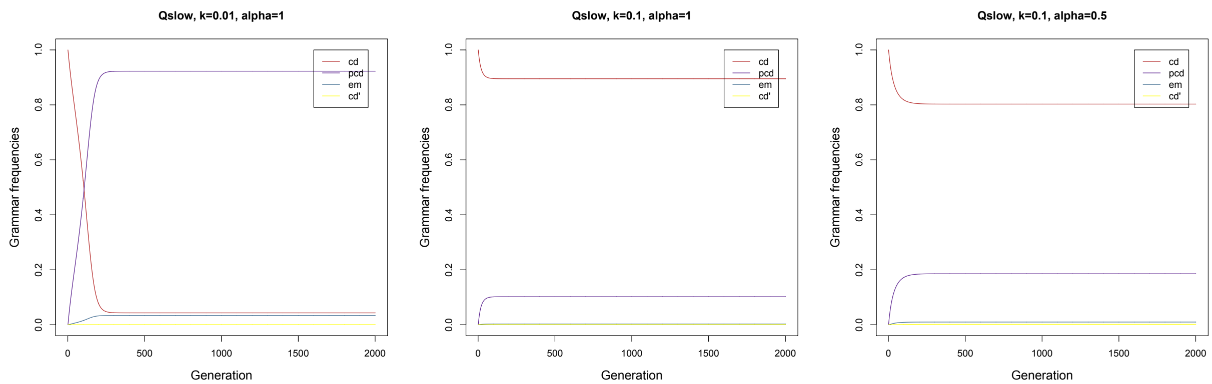



```
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,generate_A(0.1),Q,alpha=0.5), 200))
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,generate_A(0.1),Q,alpha=1), 200))
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,generate_A(0.1),Q,alpha=2), 200))
```

So even though by varying α we can create different internal equilibria, we cannot achieve a nearly-full establishment of some middle strategy followed by a takeover by *cd′*.

Another tool at our disposal is varying the rate of out-mutations. If we lower that rate, the functional considerations modelled by the fitness step of the RM dynamics will become relatively stronger. For example, we can use Q_{slow} from (17), where out-mutations are 10 times slower than in the original Q. This tips the balance heavily towards fitness, and as we can see in (33), it becomes much easier to get stuck in local equilibria. But the fundamental problem of deriving both going through to an all-cd' stage and establishing middle grammars *pcd* and *em* will not be solved.

(33) **Ev ImpGame with slower mutations as defined in Q_{slow}:**




```
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,generate_A(0.01),Q_slow), 2000))
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,generate_A(0.1),Q_slow), 2000))
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,generate_A(0.1),Q_slow,alpha=0.5), 2000))
```

4.4. **Summary: infinite-population replicator-mutator dynamics for the progressive-imperfective cycle.** We have examined how the infinite-population RM dynamics behaves under (a) the fitness matrix A with different costs k factored in, (b) different values of α determining the relative size of the fitness bonus, and (c) different out-mutation rates in Q and Q_{slow}. Under this wide range of conditions, we observed that the two desiderata for the progressive-imperfective cycle, (i) that the cycle goes through to a final all-cd' stage, and (ii) that the cycle passes through stages where the middle grammars *pcd* and *em* are close to 100% in the population, — seem incompatible. If we go through to the final cd' stage, we never get close to the full establishment of *pcd* and *em*. If, on the other hand, we get one of the middle grammars established, this will be at the expense of never reaching an all-cd' stage.

Given the evolutionary model we used, this should not be very surprising. First, the RM evolutionary dynamics is deterministic. Thus even in a system with multiple equilibria, the initial state will fully determine which equilibrium the system will end up in. There is no possibility of going to one equilibrium, spending a little time in it, but then leaving it to travel to another one. Furthermore, the structure of the NF ImpGame is such that establishing one of the middle grammars entails being stuck in a stable equilibrium. Indeed, let's ask ourselves what conditions are needed for the evolutionary process to reach a population where x_{pcd} is close to 100%. Consider for concreteness some specific value for x_{pcd}, such as 0.8. If x_{pcd} is to go closer to 1 from there, then at the next step, its share should increase. The out-mutation rate for *pcd* is quite high in Q, namely 9% of the *pcd* share after the fitness step. The in-mutations that go into the other directions cannot be too significant, since the shares of all other strategies only sum up to 0.2. So the in-mutations into *pcd* cannot really be higher than 2% of the overall population size, and likely will be much smaller. Consequently, if x_{pcd} is to increase further towards 100%, it needs to get a very significant boost at the fitness step — let's say, on the level of 8% of total population size. Suppose it indeed receives such a boost. But now consider what happens at the next step, where $x_{pcd} > 0.8$. The high fitness boost at the preceding step is only possible if *pcd* plays very much better against itself than any other present strategy — since x_{pcd} was at 0.8, the interactions with *pcd* were the heaviest component of each strategy's fitness score. But now x_{pcd} is even higher, and therefore *pcd* should receive an even greater fitness boost! Of course, the number of out-mutated agents will also increase simply because the overall share x_{pcd} has increased, as it is defined as a constant percentage of the current population share. At some point, we may well hit an equilibrium point where the fitness growth of x_{pcd} will be exactly compensated by out-mutation. But the bottom line is, as we move closer towards a pure *pcd* population, the relative fitness of *pcd* against the fitness of other strategies will only increase. There will never be fitness considerations pushing the system away from a pure-*pcd* state.

This simple argument shows that it is no accident that we could not derive both (i) and (ii) desiderata after trying many possible sets of parameters. This is in fact simply impossible in the infinite-population RM-dynamics setting. However, as we will see in the next section, switching to a *finite* population size will be able to change this.

5. Working with finite populations

5.1. **Finite-population replicator-mutator dynamics, and its linguistic benefits over the infinite-population one.** Setting at first aside the concern about the linguistic plausibility of the infinite-population assumption, let's review the fundamental mathematical difference between working in the infinite and finite population settings. Both leaving offspring (the fitness step) and mutating (the mutation step) are supposed to be probabilistic processes. The ratio $f(\bar{x})/\phi(\bar{x})$ models the *expected* number of offspring, and the mutation rate Q_{ij} provides the *probability* that the offspring of an i parent will acquire grammar j. But when our population is infinite, those average, probabilistic notions turn into deterministic ones. Indeed, if we toss a fair coin a couple of times, by chance we can get very different outcomes.

But if we could toss it an infinite number of times, it would have landed tails exactly 50% of the time. All individual probabilistic events are averaged out in the infinity. Therefore the evolution dynamics for the infinite-population case is deterministic: the actual new shares are exactly the expected new shares.

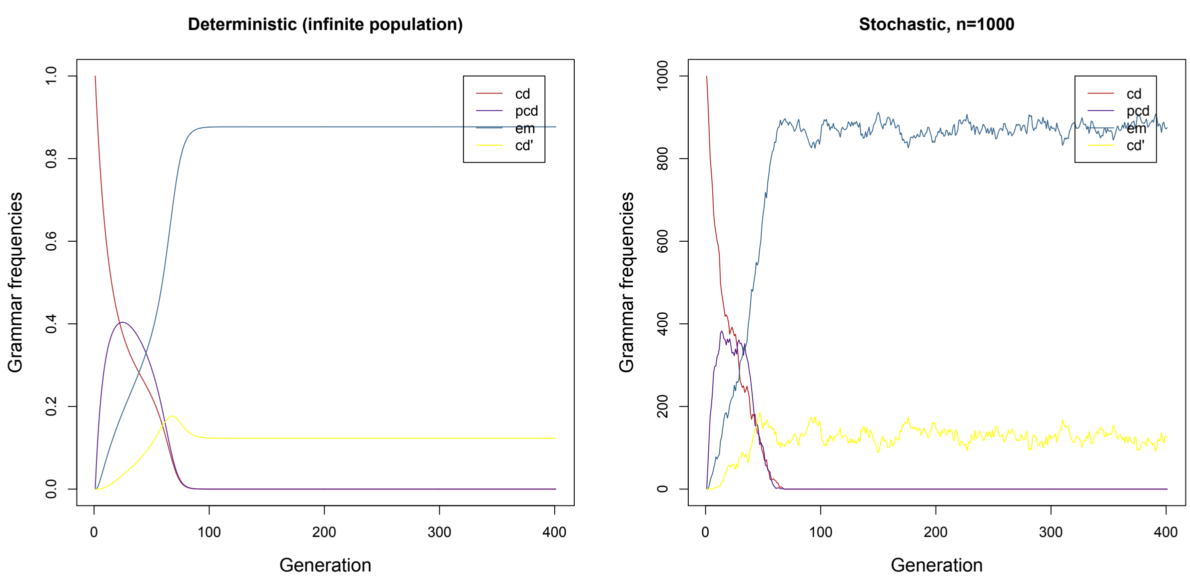
But this is not so in finite populations. Suppose that 100 individual *cd* speakers have the probability 0.06 of turning into *pcd* speakers. Since for each speaker, the grammar switch or its absence is independent from what happens to the other speakers, we will get a different number of switchers in different trials. Sometimes exactly six, the expected number, will switch. But sometimes it will be 2, or 7, or 11, or 0. And the same goes for the number of offspring. If a *cd* speaker is expected to have 1.1 offspring on average, that means that some of them will have 1, some 2, and some others might have 0, or 3, or even more. The greater the size of the population, the closer the overall number of such offspring will be to $1.1x_{cd}$. For example, with one million agents, the deviation from the average will be usually very small compared to the population size. But with 10, 25 or 100 speakers, it may be quite significant. Importantly, each new trial can produce a different result. So the population state at the next step cannot be deterministically deduced from the current population state. Instead, the evolutionary dynamics in the finite case defines the *probabilities* for a state $\bar{x}$ shifting into a state $\bar{x}'$. Evolution becomes stochastic, and different simulations with the same starting conditions will produce different results.

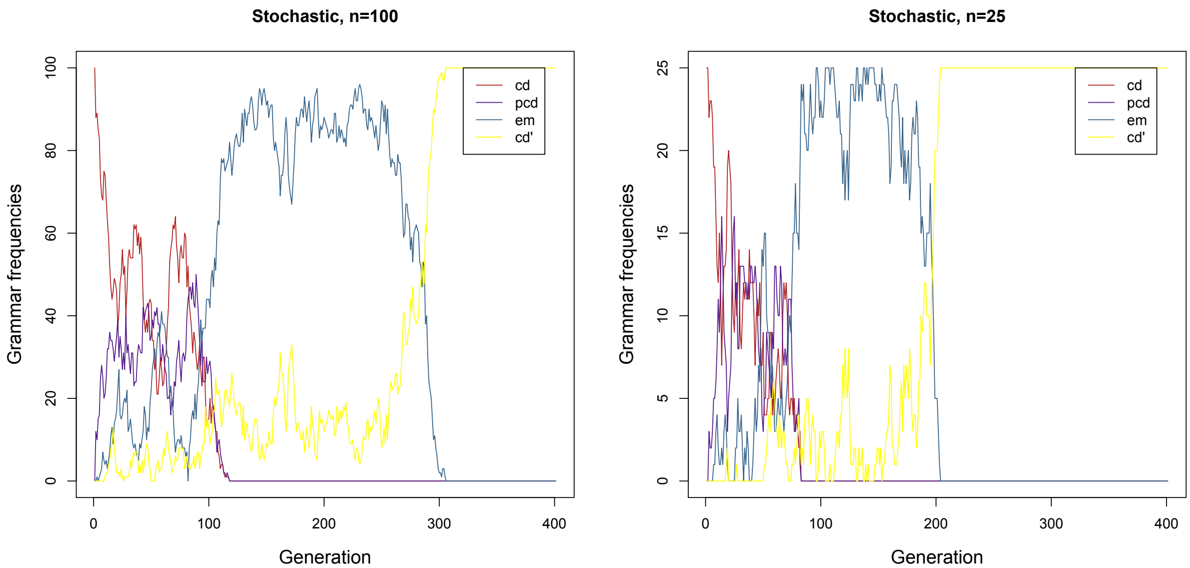
Besides the fact that the finite-population setting is theoretically more appealing (after all, human languages are not spoken by infinite populations!), there is also an important modeling benefit to it. Our problem in the previous section was that when we could get close to establishing one of the middle grammars *pcd* and *em*, this entailed being stuck at that near-establishment stage, because of the local stable equilibrium. The stochasticity of the finite-population evolutionary process means that we may become able to un-stuck ourselves. Because of random fluctuations due to chance, sometimes the system would jump far enough from the local equilibrium to cross into the neighboring basin of attraction. In other words, the stochastic nature of finite-population evolution may create the possibility of passing through several stable equilibria over the course of the cycle.

Let us illustrate this benefit immediately. For that, we define a stochastic version of the RM dynamics. In the finite setting, the shares of all grammars in the population should be integers, and sum up to n, the size of our population. For simplicity, let's keep the population size constant. Each next step in the evolutionary process is defined as follows. First, n offspring are borne. For each strategy i, the probability that a new agent is born by an i-parent is exactly x_i times $f_i(\bar{x})/\phi(\bar{x})$. In other words, the expected number of offspring for an individual i-agent is $f_i(\bar{x})/\phi(\bar{x})$. This is the fitness step. Second, for each offspring born by an i-parent, we determine which grammar they will actually adopt, based on the probabilities in the i-th row of Q. This is the mutation step. It is easy to see that this stochastic model is an exact parallel of the replicator-mutator discrete-time dynamics.

We can now compare the deterministic and stochastic models, setting n to 1000, 100 and 25. Example diagrams are given in (34). (NB: the stochastic trajectories will not be reproduced exactly when re-running the same command; though the general properties of the evolutionary process are preserved between different runs, the precise events in each simulation cannot be predicted exactly because the process is stochastic.) With $n = 1000$, the evolution trajectories of the stochastic process are quite close to the deterministic trajectories. The process gets stuck in the internal equilibrium between *em* and cd'. But when we change n to 100, we can suddenly observe that after spending some time with x_{em} above 75%, the process switches to an all-cd' population. With $n = 25$, the jumps in the process become even more erratic, and x_{em} reaches the full 100% several times, before the process crosses over to reach the final all-cd' stage.

(34) **Deterministic vs. stochastic replicator-mutator evolution:**





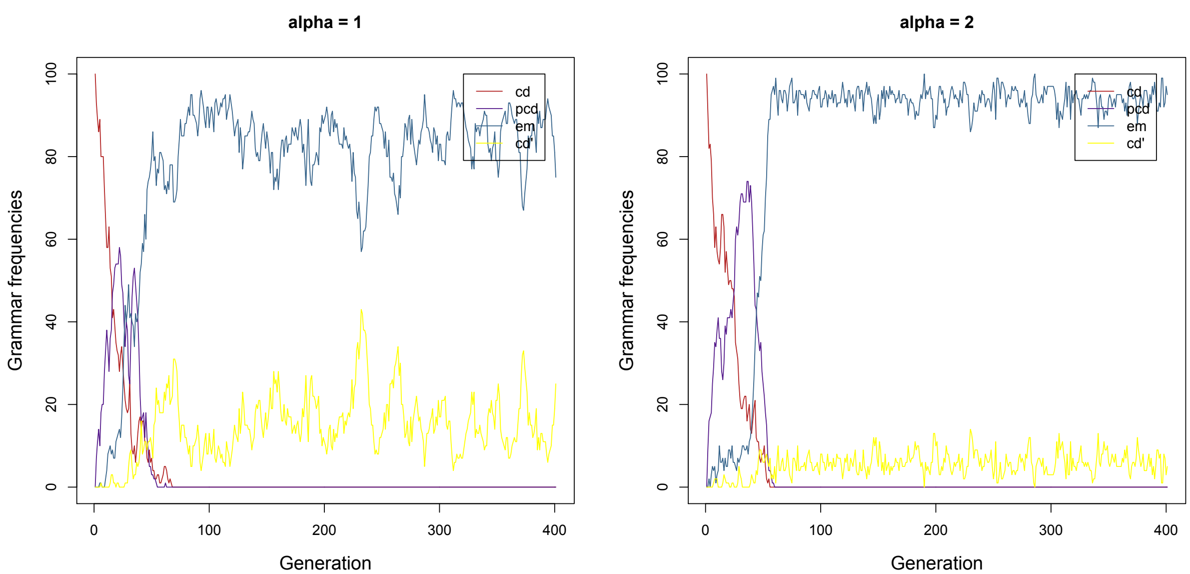

```
plot_imperfective_share(run_n_generations(x_start, function(x)
    replicator_mutator_discrete(x,generate_A(0.01),Q), 400),
plot_imperfective_share(run_n_generations(c(1000,0,0,0), function(x)
    stochastic_RM_ReplaceAll_ConstantSize(x,generate_A(0.01),Q), 400))
plot_imperfective_share(run_n_generations(c(100,0,0,0), function(x)
    stochastic_RM_ReplaceAll_ConstantSize(x,generate_A(0.01),Q), 400))
plot_imperfective_share(run_n_generations(c(25,0,0,0), function(x)
    stochastic_RM_ReplaceAll_ConstantSize(x,generate_A(0.01),Q), 400))
```

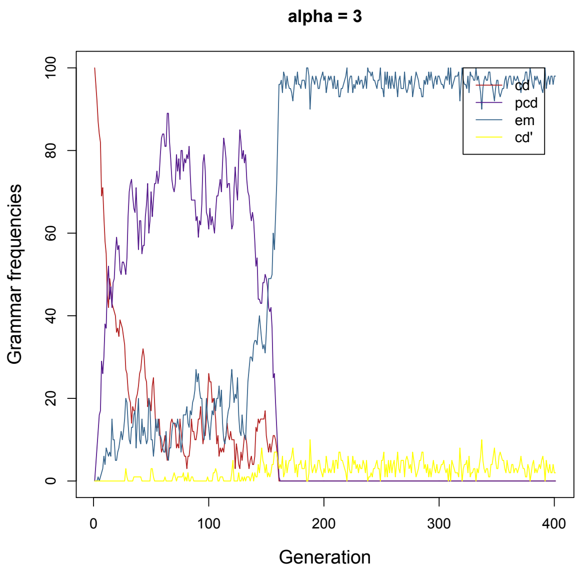
Quite generally, stochastic evolution favors uniform populations at lower n-s. This is a direct consequence of the probabilistic properties of the process. Consider the fate of a 10% share of cd' speakers in a majority *em* population at $n = 100$ and $n = 10$. At $n = 100$, the 10% share consists of 10 individuals, and chances are, at least one of them, and likely more, will leave some offspring. But at $n = 10$, a 10% share is just 1 individual. The chance that a single individual will remain childless is much greater than that 10 individuals would. So we will generally see uniform populations more often at lower population sizes. In biology, the force that favors uniformity because of the sheer mathematics of random reproduction is called *genetic drift*. It is the third force, together with fitness and mutation, that determines how evolutionary processes work.

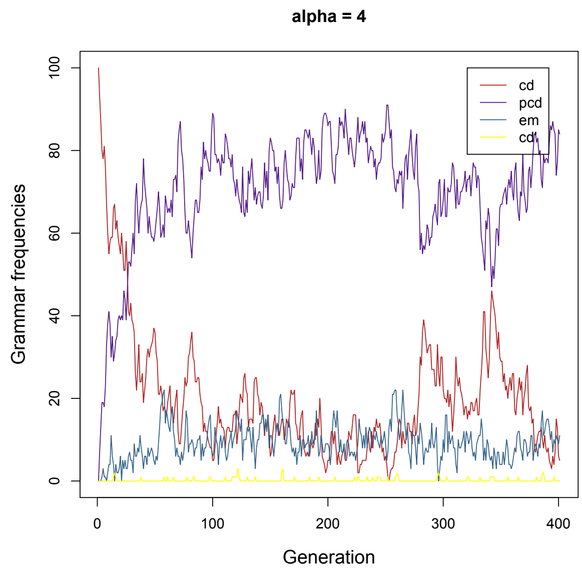
We can congratulate ourselves: just by switching to the finite setting, we have already been able to derive desideratum (i) and one half of desideratum (ii): at $n = 100$ and lower, we not only get through to an all-cd' population that models the final stage of the cycle, but also have a period where grammar *em* is nearly universal in the population. If only we could derive such a stage for the other middle grammar, *pcd*! But that stronger goal, the establishment of both *em* and *pcd*, is not easy to get. Let us try to reach this goal by using what we learned in the previous section about the general behavior of the Evolutionary Imperfective Game under the RM dynamics. Through doing that, we will see how understanding the deterministic, infinite-population dynamics may prove handy in the stochastic, finite-population setting.

In (34), we do not get close to a full establishment of *pcd* — instead, just as in the deterministic dynamics, we very quickly pass through an initial stage where *pcd* is at some noticeable level to one where *em* achieves dominance. There are two differences between *pcd* and *em* which could in principle be the reason for that. First, mutations favor fast transfer towards the later strategies. Second, the fitness of *em* against *em* is better than the fitness of *pcd* against *pcd*. In fact, it is the first factor which is more significant for the outcomes that we got above. We can counteract it by tipping the balance of our evolutionary process towards fitness — for example, by increasing α that determines the size of the fitness bonus to reproduction:

(35) **Better establishment of *pcd* with increasing α (Deo's Q, A with $k = 0.01$, $n = 100$):**





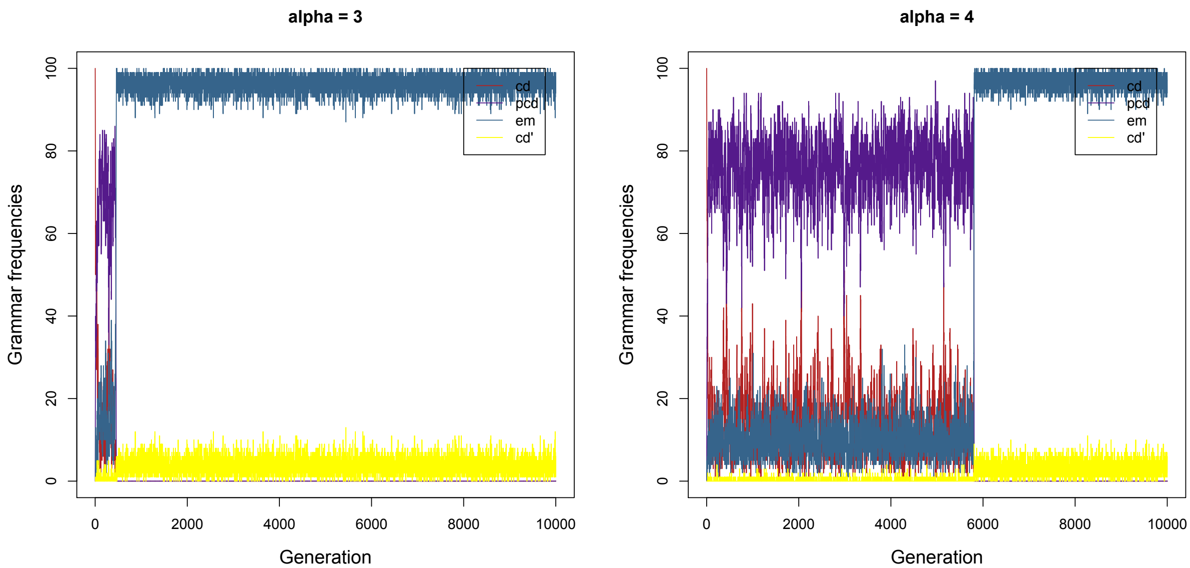



```
plot_imperfective_share(run_n_generations(c(100,0,0,0), function(x)
    stochastic_RM_ReplaceAll_ConstantSize(x,generate_A(0.01),Q,alpha=1), 400))
plot_imperfective_share(run_n_generations(c(100,0,0,0), function(x)
    stochastic_RM_ReplaceAll_ConstantSize(x,generate_A(0.01),Q,alpha=2), 400))
plot_imperfective_share(run_n_generations(c(100,0,0,0), function(x)
    stochastic_RM_ReplaceAll_ConstantSize(x,generate_A(0.01),Q,alpha=3), 400))
plot_imperfective_share(run_n_generations(c(100,0,0,0), function(x)
    stochastic_RM_ReplaceAll_ConstantSize(x,generate_A(0.01),Q,alpha=4), 400))
```

When we increased α with the same Q and A in the deterministic setting, we obtained *pcd* peaking over 50% at $\alpha = 2$, and a stable equilibrium with $x_{pcd} > 60\%$ at $\alpha = 3$, see (31) above. In the stochastic case, $\alpha = 2$ also drags *pcd*'s peak higher, but does not help it stay there. With $\alpha = 3$, we get x_{pcd} hovering around 70% of the population for a bit. With $\alpha = 4$, we get a *pcd*-dominance phase which does not disappear over 400 generations.

But it is too early to rejoice. Let's run the same simulations that derive a stable *pcd*-dominant state for a longer time, to see the longer-term behavior:

(36) **Longer-term behavior of trajectories with established *pcd* dominance (Deo's Q, A with $k = 0.01$, $n = 100$):**

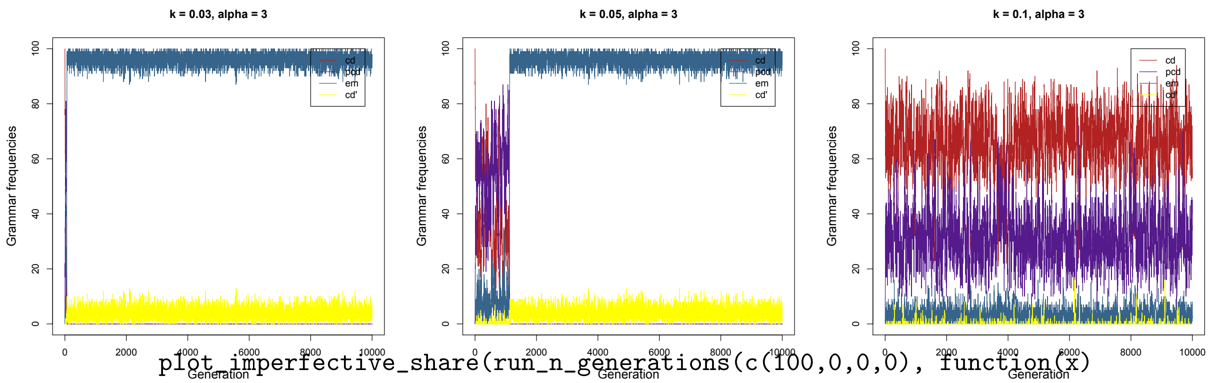



```
plot_imperfective_share(run_n_generations(c(100,0,0,0), function(x)
    stochastic_RM_ReplaceAll_ConstantSize(x,generate_A(0.01),Q,alpha=3), 10000))
plot_imperfective_share(run_n_generations(c(100,0,0,0), function(x)
    stochastic_RM_ReplaceAll_ConstantSize(x,generate_A(0.01),Q,alpha=4), 10000))
```

We are successful in establishing both a *pcd* stage and an *em* stage. But we do not get to a cd' stage! (In a super-long time horizon, such a switch can eventually happen, but several tens of thousands generations usually will not be enough for that.) This is so because with $k = 0.01$ and high α, the fitness benefit of *em* becomes so high relative to cd' that the random events which could have led to a switch become very improbable.

We can try to improve on that by increasing the cost k, and thus diminishing the relative fitness of *em* compared to cd'. But first, increasing k also makes *pcd*'s position weaker, so we do not derive a nice *pcd*-dominant stage; and second, this is still not enough to create the conditions for a cd' takeover. In fact, at some point we just stop deriving an invasion of *cd* into *pcd*, as shown below in the condition $k = 0.1$ on the rightmost diagram:

(37) **Increasing k does not lead to a takeover by cd' at the end stage (Deo's Q, $n = 100$, k = 0.03, 0.05, 0.1):**

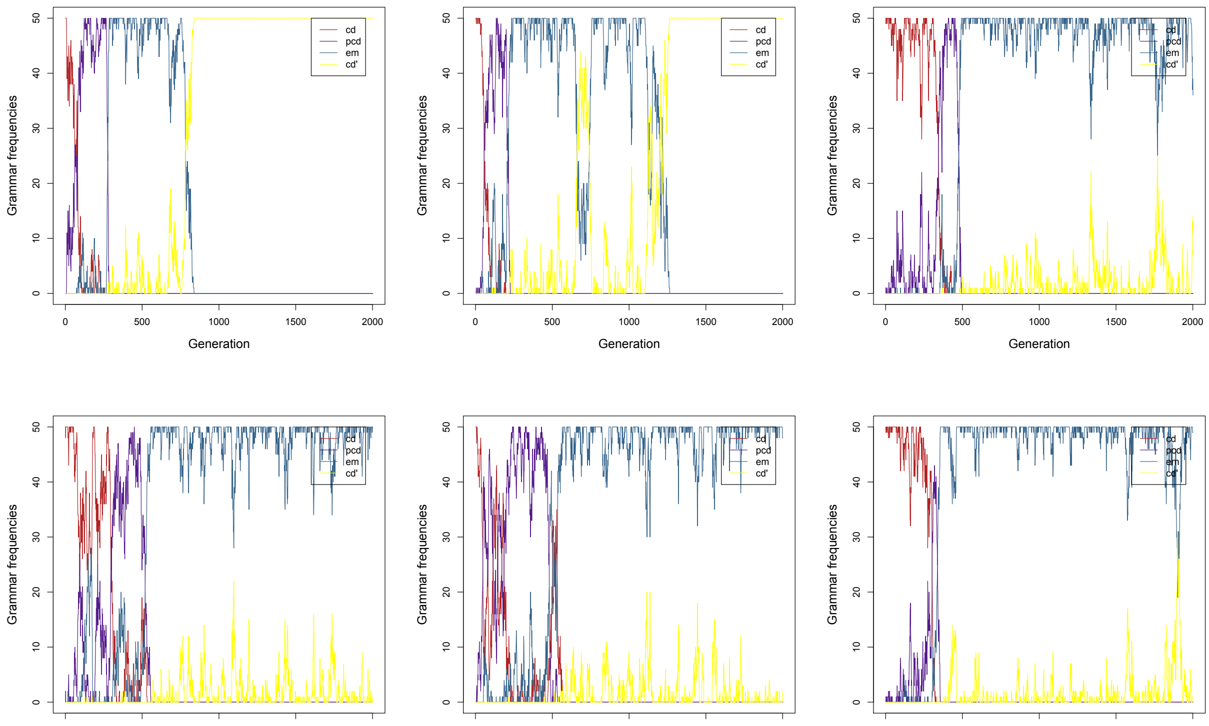



```
plot_imperfective_share(run_n_generations(c(100,0,0,0), function(x)
    stochastic_RM_ReplaceAll_ConstantSize(x,generate_A(0.03),Q,alpha=3), 10000))
plot_imperfective_share(run_n_generations(c(100,0,0,0), function(x)
    stochastic_RM_ReplaceAll_ConstantSize(x,generate_A(0.05),Q,alpha=3), 10000))
plot_imperfective_share(run_n_generations(c(100,0,0,0), function(x)
    stochastic_RM_ReplaceAll_ConstantSize(x,generate_A(0.1),Q,alpha=3), 10000))
```

Another option which can boost uniform population states while not disadvantaging too much movement through the cycle is reducing the size of our population. With a smaller population, jumps between relatively stable states become more probable. Moreover, genetic drift, which favors uniformity, also becomes stronger.

Indeed, if we lower n somewhat, we can derive evolutionary trajectories that are more or less resemblant of the desired ones. But since a smaller population size also means more stochasticity, simulation runs with such smaller populations will be more unlike one another. Moreover, since the role of randomness is increased, they will allow more random jumps that are not guided by the forces of fitness and mutation, and thus sometimes override the effect of those. We illustrate for $n = 50$, $k = 0.03$, and slow mutations in Q_{slow}:

(38) **Several runs with identical starting conditions $n = 50$, Q_{slow}, A with $k = 0.03$, 2000 generations each:**




```
plot_imperfective_share(run_n_generations(c(50,0,0,0), function(x)
    stochastic_RM_ReplaceAll_ConstantSize(x,generate_A(0.03),Q_slow), 2000))
```

The first two diagrams in (38) support our desiderata (i) and (ii): each of the middle grammars achieves nearly 100% dominance, even if for a short time, and the cycle goes through to the final stage. Note that in the second diagram, we can see an almost complete takeover by *cd′* which then bounces back to *em* dominance again. Such reversals are an expected effect of increased randomness in our process.

The third, fourth and fifth diagrams also show neat establishment of *pcd* and *em* stages, and in the third and the fourth, we also have an initial *cd* stage. That we do not see yet a *cd′* takeover is not a reason for concern here: if we ran these simulations longer, sooner or later we would have gotten it, as we did in the first two diagrams.

But then the last diagram shows an initial *cd* stage which very quickly turns into an *em* stage without really passing through *pcd*. In other words, our computational model of the progressive-imperfective cycle sometimes skips one of the middle stages!

Let us sum up. Depending on the parameters, we are able to achieve one of the following three results in (39):

(39) **Possible outcomes for stochastic replicator-mutator modeling in Ev ImpGame:**

(1) No *pcd* stage; a clear *em* stage with an admixture of 10-20% *cd*′; a *cd*′ takeover.

E.g., $n = 100$, $\alpha = 1$, Deo's Q and A with $k = 0.01$, cf. the third diagram in (34).

(2) Clear *pcd* and *em* stages (with *cd*+*em* and *cd*′ admixtures, respectively); a *cd*′ takeover very improbable.

E.g., $n = 100$, $\alpha = 3$ or 4, Deo's Q and A with $k = 0.01$, cf. the diagrams in (36).

(3) Good sequences of *pcd*, *em* and *cd*′ stages on many runs, but the behavior is generally not very robust between runs, and two important deviations from the *a priori* desired trajectories emerge: (a) sometimes the *pcd* stage goes missing, and (b) generally, the probability of back-developments (for example, retreat from 75% of *cd*′ back to an all-*em* population) is significant.

E.g., $n = 50$, $\alpha = 0.3$, Q_{slow} and A with $k = 0.03$, cf. the diagrams in (38).

5.2. **Linguistic interpretation of stochastic replicator-mutator results.** In the deterministic, infinite population setting, we could not derive the desiderata (i) and (ii) together. In the stochastic case, we can do it under some settings, but there are still features in our simulations that differ from what we *a priori* considered desired for the modeling of the progressive-imperfective cycle. We summarized the three main types of partially welcome modeling outcomes in (39).

Now we come to an important analytic step: we need to interpret the deviations of our predictions from the *a priori* expected trajectories. Interestingly, our predictions in (39) do not look completely implausible. In fact, they make salient the fact that we do not currently have numerical temporal data for the actual progressive-imperfective cycles in language change. We discuss the outcomes, and the empirical predictions they make, one by one. Note that below we crucially do not use the absolute number of generations in our simulation runs as an argument. This is so because the interpretation of that number may vary — it need not be equal to the number of human generations (as [Deo, 2015] interprets that number.) If it were, then having the cycle run over 2000 generations or more would be too long: this would translate to about 40 thousand years. Luckily, there is another interpretation for the number of generations, under which they may be much shorter. We will return to that in the next section. However, even without appealing to the absolute length of the stages, we can and should appeal to their *relative* length. A stage that is on average 10 times longer in simulated generations than some other stage, will remain so whatever our interpretation of a single generation might be.

Consider outcome (1) first. In this outcome, no *pcd* stage arises. Furthermore, at the equilibrium the *em* stage with categorical use of form X for the imperfective and form Y for the progressive has a significant admixture of agents using only the newer Y form in all circumstances. In my judgement, this is the least plausible of the three possibilities in (39). The non-existence, or an extremely short duration, of the *pcd* stage is a serious problem. As we discussed in fn. (1), the several centuries of the Early Modern English period is the time when the progressive "be V.ing" already existed, but was not yet obligatory (one might remember Shakespeare's Polonius's "What do you read, my lord?", which clearly has the progressive meaning). The simple present ceased to be able to denote progressive meanings only in the 19th century. On the face of it at least, it looks as if not deriving a *pcd* stage is a very bad outcome. Furthermore, an *em* stage with an obligatory admixture of *cd′* agents seems implausible as well. There *are* languages that have a nearly complementary distribution of progressive and imperfective forms: in fact, Present-Day English very much works like that. And it is not that there were speakers of English in the 19th century who only used the progressive form, and never, for example, the simple present. The trajectories of outcome (1) are thus not plausible, at least for some known languages.

Turning to outcome (2), it is a nice feature that the *pcd* and *em* stages are derived there, but what to do with the absence of a *cd′* takeover? In fact, this *could* actually prove to be a beneficial feature of this outcome, depending on empirical data. In principle, there are two logical possibilities for the last shift in the progressive-imperfective cycle. First, the takeover by *cd′* may be already predestined to happen relatively shortly once the categorical *em* stage is reached. This is a possibility not compatible with modeling outcome (2). But second, it could be that a categorical system with one form for progressive meanings and another one for imperfective meanings is *internally stable*. When it receives an external push of some sort, it would be able to turn into an all-*cd′* system. But it would crucially require such a push, and would not normally move in this direction for internal reasons. This possibility corresponds perfectly to outcome (2).

Indeed, recall the reason why *em* is so stable in the sets of trajectories belonging to outcome (2). That is so because at lower k, a two-form system fully disambiguating progressive and imperfective messages is communicatively more efficient. This greater efficiency is captured in Deo's basic ImpGame and consequently NF ImpGame. It *could* be that such functionally efficient systems remain stable until they get some other external impulse (for example, changes in the rest of the tense-aspect system of a language, or gradual loss of a differentiated old imperfective form due to processes of phonological reduction, or something else).

Our modeling thus leads us to an interesting empirical question: how stable are categorical progressive-imperfective systems in the languages of the world? Can they remain in place for many centuries, or are they fated to relatively quickly give way to a single-form cd' system? At this point, we simply do not know. Large-scale empirical studies are needed to check that.

Finally, let us consider outcome (3). In this outcome, the sequence in the progressive-imperfective cycle regularly goes all the way to completion in the absence of any additional external forces. However, at least the *pcd* stage may go missing in some cases. Moreover, back-flow to an earlier stage of the cycle occurs with significant probability.

Is outcome (3) a good model of the linguistic process? Again, there is no way to tell without more data. For the *pcd* stage, we simply do not know how long it takes on average compared to the other stages of the cycle. It could be that in some languages passing through the cycle, that stage is actually passed very rapidly, just as in some of our runs. Note that the *pcd*-like stage in Early Modern English that continued for a significant time, is perfectly compatible with the predictions of outcome (3): because of great stochasticity of the process in this case, some languages are predicted to stay in the *pcd* stage for a considerable amount of time, while other may pass through it very quickly. As for the possibilities of backflow, I am not aware of empirical claims concerning backflow in the progressive-imperfective cycle, but generally examples of back evolution — or at least *seeming* examples thereof — may be found in language change. As for the current absence of positive evidence for this in the progressive-imperfective system, note that to register such a backflow event as in the second diagram of (38), we need data with high precision. The unsuccessful attempt at cd' takeover takes about 1/10 of the overall *em* stage. To have any hope of noticing something like that in actual historical data, we would need to not only know that some language A had (for example) a largely categorical progressive-imperfective system in the 12-16th centuries, but also have extensive data at least about each 50-year interval within that time period. Descriptions of such precision are not to my knowledge cited in the typological and semantic literature on the progressive-imperfective cycle.[16]

[16] Such data might exist in the philological traditions describing particular historical languages, though I am presently not aware of any such for the progressive and imperfective. But, for example, with regard to English modals, [Tellier, 1962] more or less approaches the needed level of precision, and with respect to the rise of English auxiliary *do*, [Ellegård, 1953] does.

For the English progressive at least, data of sufficient granularity are likely to come from ERC-funded project *Mind-bending grammars* which is to start in September 2015 at the University of Antwerp (PI Peter Petré). The Early Modern English progressive is one of the eight constructions that the project aims to examine, cf. `https://www.uantwerpen.be/en/projects/mind-bending-grammars/project/case-studies/`.

Summing up, though we have not derived the outcome we initially strived for, namely a clean $cd \to pcd \to em \to cd'$ sequence, our modeling nevertheless produced reasonable alternative trajectories of change, i.e. outcomes (2) and (3) above. Without new empirical research, we cannot determine whether the originally expected trajectories or those of outcomes (2) and (3) match what happens in the actual progressive-imperfective cycle. It was crucial to use the stochastic, finite-population cousin of the actual deterministic RM dynamics to derive such predictions as seem potentially plausible.

Thus the stochastic version of the RM dynamics, taken together with Deo's analysis of the progressive-imperfective interactions in NF ImpGame, is at this point a sensible competitor in the search for an evolutionary model that approximates real processes of change in the progressive-imperfective cycle. But how plausible that dynamics is when interpreted in language change terms? Above, we raised some concerns about that, and in the next section, we will finally return to this issue.

6. Evolutionary dynamics and their linguistic interpretations

So far, we have assumed the replicator-mutator dynamics, or its stochastic, finite-population modification, as our model of language change. It is time to step back and ask what commitments come together with this dynamics, and how we can change the commitments by switching to different dynamics. In this section, we will discuss two alternative interpretations of the RM dynamics, then show that one of its assumptions, namely that of single-parenthood of agents, is actually not as problematic as it might seem. Then we will examine two different evolutionary dynamics, one built from scratch starting from a set of plausible assumptions about language change, and another, the BNN-mutator dynamics that we have seen in Section 3.2. Neither of those two will work well in the Imperfective Game. We will be able to conclude that despite the initial impressions of implausibility, the RM dynamics is not necessarily that terrible when applied to language change.

6.1. Deo's linguistic interpretation for the replicator-mutator dynamics.

[Deo, 2015]'s interpretation of her evolutionary model is as follows. Agents in the model correspond to actual people; the lifetime of an agent is a person's lifetime. Each agent only has a single "linguistic parent". All offspring in the next generation are born at the same time, and do not co-exist with their parents in the population that plays the Imperfective Game. Parents with a more communicatively successful grammar for the progressive-imperfective system (as defined by A and the current population state) produce more "linguistic children" surviving into adulthood. More precisely, each parent with grammar i has on average $f_i(\bar{x})/\phi(\bar{x})$ linguistic children. Note that someone's linguistic children need not be their biological children: as [Deo, 2015] stresses, grammars are not genomes which are passed from parent to child automatically in biological reproduction, but cultural entities that are spread by social processes. In what follows, for simplicity I identify the "linguistic parent" with the caretaker, but this is my assumption, not Deo's: we can also think of some non-caretaker person having such profound influence onto the child that they end up going for that person's grammar, and not that of the caretaker(s).

The parent's grammar is copied by the child independently of which other grammars exist in the population. Children may thus be conceptualized as living with a single parent/caretaker who teaches them language in isolation from the rest of the community until parents die, children reach maturity and start to interact with other agents, and the ones who communicate more efficiently then reproduce at a higher rate. Though there is no interference from the other community members during acquisition, some number of children will inevitably misacquire the parent's grammar (according to probabilities defined in Q). The probability of misacquisition does not depend on the grammars currently present in the population (if we assume that the children never meet other people except their single parent until they read adulthood, this even makes sense). Finally, in addition to the above, for the deterministic setting we throw in the assumption that our population of agents is infinite. For the finite-population version, this assumption is lifted, yet there is an implicit assumption that communicative success of each individual depends in equal measure on the strategies of everyone else in the population. In other words, there is no social structure in our population that makes some agents interact more frequently, and we effectively assume that in the mature period, each individual has enough time to interact many times with each other agent.

It is instructive to imagine what a community of language speakers with such properties would look like. Our speakers live most of their lives in isolation — perhaps each in their own cave?.. — together with the children they raise. They are also social creatures: having become adults, they go out into the wide world and interact intensively with their kind. In fact, those interactions should very crucially involve communication, as those individuals who communicate better end up caring for more children later on. If we feel romantic, we can imagine our agents competing for caves through long rhetorical tournaments where the more eloquent ones get better living quarters, and better living quarters mean they can take care of more children (whoever their biological parents), which is a good thing in this society. If we are more prosaic, we can think of our agents communicating at, let's say, communal hunts, where better communicators somehow manage to get access to a larger share of the prey. But some way or other, communication must be very important for determining how many children they will raise, according to our model. After the social period of their lives, our creatures go back into the caves, together with the children who they will take care of until those are ready to go out. Though language will be very important for children's future social success, it should be noted that they misacquire their parents' grammars quite a lot (the out-migration rates in Deo's Q are in the range of 3% to 9%). But we as observers may note that evolutionarily, this can sometimes be a neat strategy, as such outmigrations may lead our agents towards grammars with better communicative properties (i.e. help them cross the valleys of the fitness landscape.) The inability of our agents to always acquire the progressive-imperfective grammar of their parents correctly is therefore not necessarily a hindrance, at least on the community scale.

While the picture above may seem ridiculous, it should not be taken so. Though it differs vastly from what we think language acquisition is like, it is still an internally coherent, sensible picture. There *could* be creatures living in such a setup. The real question we want to ask is not whether the model is similar in all respects to what we know about human language acquisition, but whether the model's *differences* from our assumptions are harmful for its predictions. For example, in population genetics it is common to start one's analysis with a model populated by asexual agents that cease to exist once they reproduce. Quite obviously, while reasonably applicable to single-celled organisms reproducing by division, this hardly resembles how mammals evolve. Yet studying evolution on that mathematically simpler model, we can still get important insights that would carry on to more complex cases. The RM model as applied to language change should not be discarded right away just because spelling out the consequences of Deo's interpretation for it results in implausible assumptions. But can we do better than those? Below we show that we can.

6.2. **An alternative linguistic interpretation for the replicator-mutator dynamics.** Though some of the assumptions behind Deo's interpretation of the RM dynamics are linguistically implausible, there is in fact an alternative, arguably less implausible, interpretation of the same mathematical model. We will now present that interpretation.

Deo's identification of agents in the model with actual people follows from the axiom, shared by much of the generative tradition on language change, that language change happens only through (mis)acquisition of grammar in early childhood. If the childhood is the only time you can change your grammar, then indeed your evolutionary agents should be identified with people.

But the axiom is surely not true in its strong form: language *use* definitely changes across the lifetime. This is not the place to discuss whether such change in use reflects an underlying change in grammar (as a sociolinguist would assume), or the grammar is something very fundamental and immutable, while frequencies of use reveal nothing about it (as a generative scholar could argue). What is important for the purposes of the present paper is that *if* we reject the axiom, a road opens to arguably a more natural linguistic interpretation of the RM dynamics.

Under this interpretation, an agent is a *time-slice* of an actual person. The next generation consists of the next time-slices of the present speakers. The current grammar is inherited by the current agent's offspring simply as a matter of retention: the default case is that my grammar tomorrow will be the same as my grammar today. In addition to faithful transmission, there are two types of change. First, our agent may notice that some other agent communicates more efficiently than themselves. This can make our agent to adopt that other agent's grammar. So while in the default case, the "grammatical parent" of the next time-slice of a specific individual is that individual themselves, sometimes it will be some other individual whose grammar is adopted. The rate of such adoption is given by ration $f(\bar{x})/\phi(\bar{x})$: communicators better than average tend to have their grammars adopted by other agents, and those who are worse than the average tend to borrow other people's grammars. This mechanism is responsible for the replicator, or fitness, part of the dynamics. The second way to change is when speakers spontaneously shift from the grammar that they use to another one, in accordance with the probabilities defined in Q. Unlike fitness switches, such mutation switches occur independently of the current population state. This part remains the same as in Deo's interpretation: we build into our model the assumption of constant switches between grammars. As we have seen in Section 4, mutations are absolutely crucial for deriving anything close to the progressive-imperfective cycle if Deo's NF ImpGame is on the right track. Without mutation, the fitness part will nudge populations towards one of the pure-grammar states, preventing any progression through the cycle.

In a realistic model, speakers would sometimes die, and new speakers would sometimes enter the population. In principle, these replacement processes could add another component to the dynamics. But here, we abstract away from this completely (thus keeping the RM dynamics as is). Doing this, we are effectively saying that the new individual entering the population inherits their grammar from the individual who passes away just as the latter's next time-slice would have inherited it.

Which of the two interpretations, Deo's or our alternative, is better? To a large extent this is a question of taste — or of one's theoretical convictions about language change. But I argue that in one aspect, my alternative interpretation is somewhat superior: its interpretation of fitness. In both interpretations, fitness determines the number of "linguistic children" of an agent. On Deo's interpretation, either better communicators would literally have more children, or they would be able to nudge the child towards their grammar even though the grammars of the child's caretakers are different. On our interpretation, it is adults who sometimes would copy a communicative strategy of another adult if it is superior to their own. It seems more plausible to me to assume that each successful communicator would become a role model for their peers than that she or he would have access to a child who would choose their grammar over that of their caretakers.

Another difference concerns the single-parent assumption. The assumption of Deo's interpretation that each child learns their language from a single (linguistic) parent is not very attractive intuitively. Even when there is only one primary caretaker, the child would interact with several, though possibly not very many, adults. In our alternative, the "parent" is simply the preceding time-slice of the same speaker. It is natural that *this* parent is single! But this difference, though it might seem big, is actually not very significant. In the next subsection, we will show how to defineda multiple parenthood dynamics, and observe that its general properties for Ev ImpGame seem to be the same as in the vanilla RM case. This is thus an example of when an intuitively unappealing feature of the model actually does not affect the overall results by much.

One more issue where our alternative interpretation may be superior is the treatment of time. In Deo's interpretation, speakers from different generations never communicate. In our alternative, we have another extreme: the model behaves as if our speakers never died or had a childhood. However, the latter position is arguably more justifiable, since replacements in a population may be assumed to be rare. If so, then the default case would be indeed communication between exactly the same agents as in the previous generation, though some of them may have switched grammars since then. And importantly, this agent time-slice interpretation allows us to consider processes that proceed slowly through many thousands of modeling generations. As generations for us are simply time-slices, we can assign to them different real-world temporal correlates: they may correspond to weeks, years, or tens of years. But in Deo's interpretation, we do not have such leeway, as the modeling generation is identified with a real-world human generation. So 100 generations in the model would be roughly equal to 2000-3000 years. Given what we know about the progressive-imperfective cycle (as well as other famous cycles in the grammatical system), stages in them often take several centuries, not several thousands of years, so under Deo's interpretation, the evolutionary process should better run its course over 100-200 generations at most. In the stochastic model especially, this becomes particularly inconvenient. The benefits of a stochastic model appear exactly when we can have long time-spans with many random events: each individual event is not very likely to change the overall state of the system, but if we wait long enough, we will go through a nicely defined sequence of changes. When we restrict the number of modeling generations to a small number, we lose this: in order for the randomness to play out, the probability of switches must be quite high, but then the whole evolutionary process becomes more erratic rather than reasonably monotonic-like.

Summing up, the RM dynamics that [Deo, 2015] uses, as well as its finite-population version, need not be coupled with Deo's linguistic interpretation. If one believes that language change (in the progressive-imperfective system) happens only in the childhood, then Deo's interpretation must be adopted. But if we drop that axiom, we get access to the "time-slice" interpretation of agents in our model. This alternative offers a reasonable interpretation of fitness in the linguistic context, and allows us more flexibility with how we map our "modeling generations" to real-life time.

6.3. **Accommodating many parents.** Here, we introduce two dynamics that depart minimally from the RM, but assign several "linguistic parents" (for instance, two) to each new agent. Studying those will show that adopting the single-parent assumption is actually not such a terrible move as it might seem, as it has only limited effect on the model's behavior.

In the single-parent model, each agent has one individual whose grammar they would inherit in the default case. When there are two parents, let's say a_i speaking i and a_j speaking j, we need to provide a rule that says which grammar their child b is going to have. Now if this choice is done simply at random, our multiple-parent model mathematically collapses to the single-parent one. Indeed, the children of all i-j parent pairs will then be divided equally into i-children and j-children. Assuming that each pair produces exactly two offspring, for the i-j pair one of them will speak i, and the other j. We will have the same law of inheritance as in the single-parent case.

So let's look at more interesting cases. One possibility is to assume that parent a_i is $f_i(\bar{x})/f_j(\bar{x})$ more likely than a_j to pass their grammar on to child b. In words, a more communicatively successful parent would have better chances to pass on their grammar. This is a plausible model: after all, children could be assumed to have had ample time to examine their parents' communicative efficiency, and if so, it's reasonable for them to be biased towards copying the more efficient one. As with the original choice of the replicator-mutator dynamics, the choice of the f_i/f_j ratio is arbitrary: we could have chosen some other formula that connects f_i, f_j, and potentially $\phi(\bar{x})$ as well. But let's see what happens with this choice, assuming that 1) there are no further fitness considerations (unlike in Deo's model), and 2) the mutator part of the dynamics remains the same. Assuming an infinite population and therefore a deterministic dynamics, the formula describing this model will be as in (40).[17]

[17] Here is how we arrive at (40). If all members of the population pair up, there will be $x_i x_j$ number of i-j pairs. Since the population size is constant according to our assumptions (after all, it is infinite), each parent would (on average) reproduce twice. So there will be exactly $2x_i x_j$ children with an i-j parent pair. Now, those $2x_i x_j$ offspring will be f_i/f_j more likely to inherit i rather than j. This means that they have probability $\frac{f_i}{f_i+f_j}$ of inheriting i, and $\frac{f_j}{f_i+f_j}$ of becoming j. Furthermore, at the mutator step those individuals who inherited i have Q_{ik} chances of ending up

(40) **Two-parent fitness-ratio mutator discrete-time dynamics:**

$$x'_k = \sum_i x_i Q_{ik} \sum_j x_j \frac{f_i(\bar{x})}{2f_i(\bar{x}) + f_j(\bar{x})}$$

We can compare this new dynamics to the RM one. What we have done is essentially replace the $\frac{f_i(\bar{x})}{\phi(\bar{x})}$ term of the RM dynamics with term $\sum_j x_j \frac{2f_i(\bar{x})}{f_i(\bar{x})+f_j(\bar{x})}$. Of course, this change results in different evolutionary trajectories. For example, at the moment I do not know whether the new dynamics in (40) can reproduce the trajectories in the third diagram in (26), where grammars *pcd* and *em* never overtake the initial *cd*. But this would not be the kind of trajectory that we'd like to have for modeling the progressive-imperfective cycle anyway. And as for trajectories that we might care more about, our many-parent dynamics fares similarly to the RM one:[18]

speaking k. So the new share of k will be a sum of such switchers into k from every grammar i. We can write down the following:

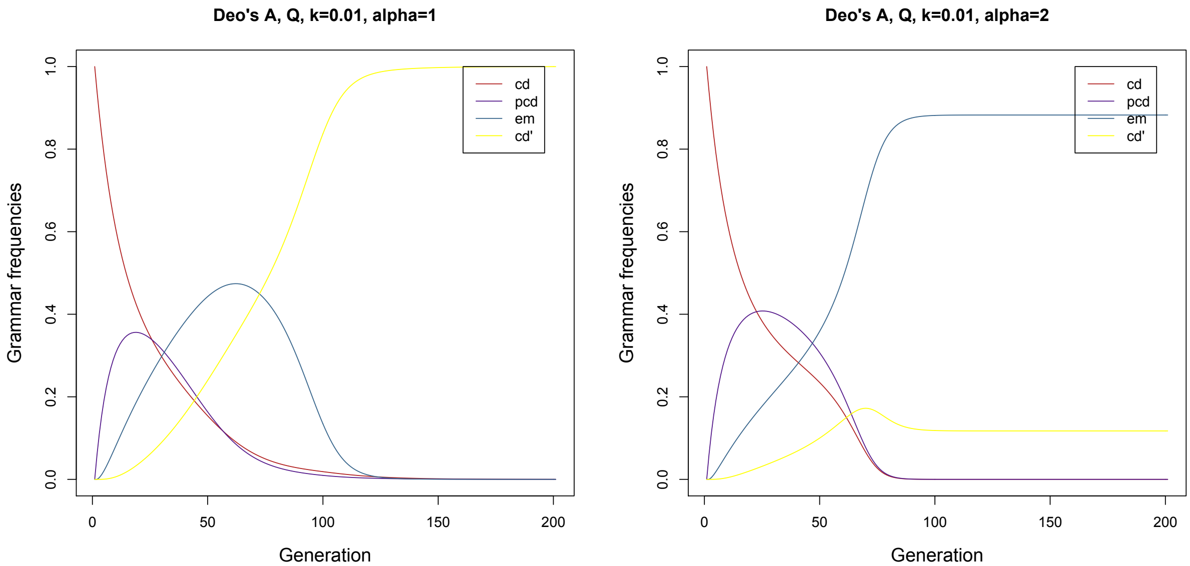
$$x'_k = \sum_i \sum_j 2x_i x_j \frac{f_i(\bar{x})}{f_i(\bar{x}) + f_j(\bar{x})} Q_{ik}$$

Since x_i and Q_{ik} do not feature j, we can move them out of $\sum_j$, which results in (40).

It is easy to see from this how to generalize to the more-than-two-parents case. For $n+1$ parents we would have $n+1$ instead of 2 because we'll need that many children to be born to those $n+1$ parents. Instead of summing over j we would sum over tuples $\langle j_1, ..., j_n \rangle$, with the term under the sum becoming $x_{j_1}...x_{j_n} \frac{f_i}{f_i + f_{j_1} + ... + f_{j_n}}$. Thus the likelihood of each parent to pass on their grammar would be proportional to their fitness, and depend on the fitnesses of the other parents.

[18]As before, we introduce parameter α that regulates how much effect fitness has. For this dynamics, α scales up the distance between f_i and the mean fitness of the population: $f_i^{adj} =^{def} max(0, mean(f_1, ..., f_n) - \alpha(mean(f_1, ..., f_n) - f_i))$. The max operator ensures that our adjusted fitness remains non-negative. This correction only becomes relevant at high values of α.

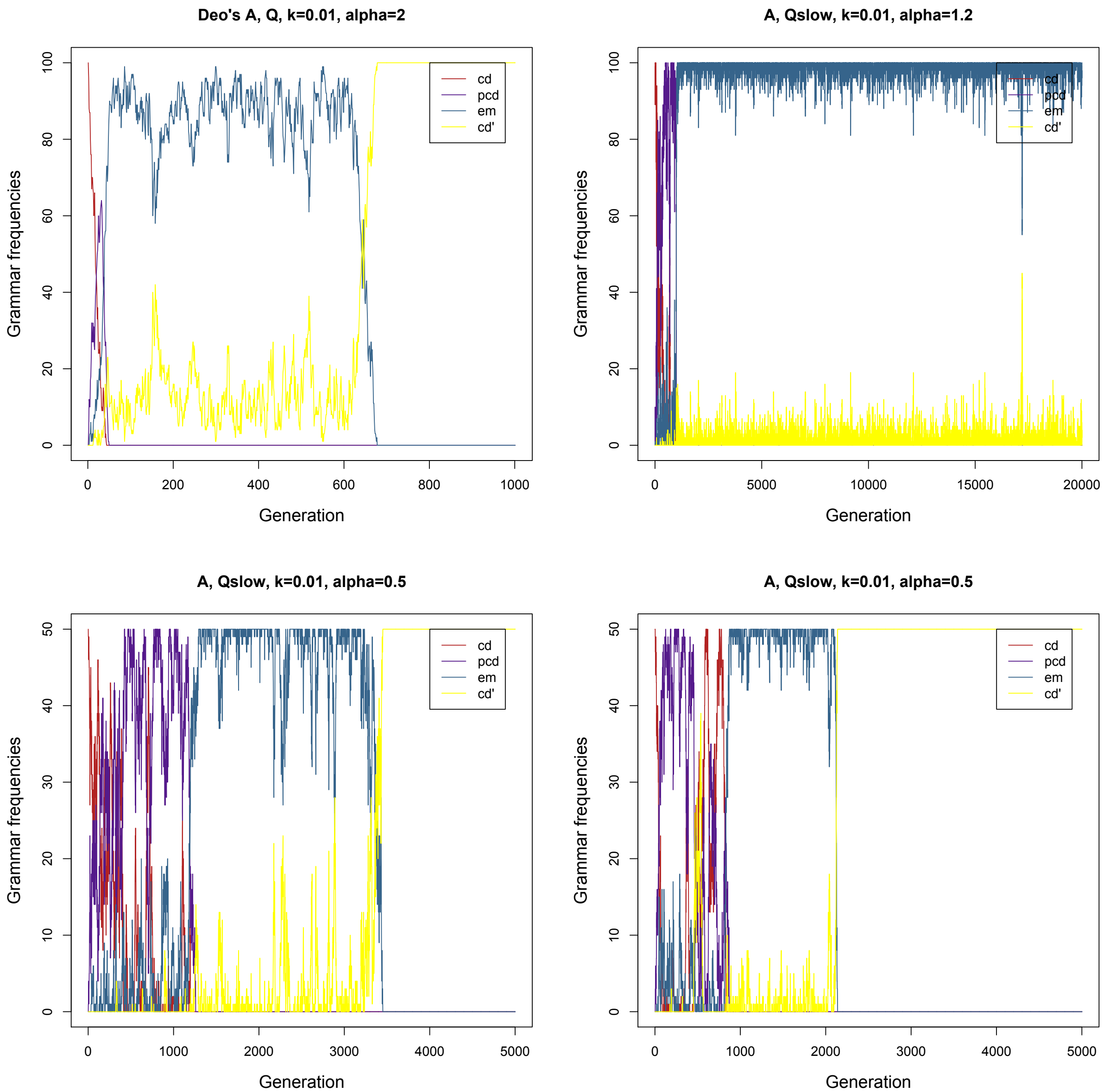
(41) **Possible trajectories under the many-parent fitness-ratio mutator dynamics:**




```
plot_imperfective_share(run_n_generations(x_start, function(x)
    two_parent_f_ratio_discrete(x,A,Q), 200))
plot_imperfective_share(run_n_generations(x_start, function(x)
    two_parent_f_ratio_discrete(x,A,Q,alpha=2), 200))
```

If we switch to the finite-population setting, the results again are quite similar: we can reproduce the three reasonable scenarios in (39), though they arise under different parameter values than in the RM dynamics. The first digram in (42) reproduces outcome (1) from (39) where there is no *pcd* stage, but an *em* plateau gives way to a *cd′* takeover. The second diagram (note the different number of generations) shows outcome (2), with *pcd* and *em* stages, but the *cd′* takeover extremely improbable. The last pair of diagrams shows that with a smaller n, we can get all three *pcd*, *em* and *cd′* stages (the third diagram of (42)), but in return the cycle's behavior becomes erratic, so that it can jump back or even skip stages (the fourth diagram of (42)). In other words, both good and bad points from our modeling results in the replicator-mutator setting carried over into the many-parents setting we just defined.

(42) **Reproducing the outcomes from (39) under finite-population many-parent fitness-ratio mutator dynamics:**




```
plot_imperfective_share(run_n_generations(x_start_100,
    function(x) stochastic_many_parents_f_ratio(x,A,Q,alpha=2), 1000))
plot_imperfective_share(run_n_generations(x_start_100,
    function(x) stochastic_many_parents_f_ratio(x,A,Q_slow,alpha=1.2), 20000))
plot_imperfective_share(run_n_generations(c(50,0,0,0),
    function(x) stochastic_many_parents_f_ratio(x,A,Q_slow,alpha=0.5), 5000)
```

In the dynamics we just studied, we assumed that each agent's rate of pairing up to produce offspring is the same, i.e. not dependent on their fitness. The fitness of an agent still played a role, but only in determining how likely they would be to pass on their grammar relative to their co-parent. Of course, there is no reason why we couldn't modify our new dynamics so that individuals with greater fitness were also more likely to get into a reproducing pair. If our previous dynamics in (40) *replaced* the fitness term of the RM dynamics, the modified dynamics will *supplement* the RM with the many-parents component. This results in the following:[19]

(43) **Two-parent fitness-ratio fitness-pairing mutator discrete-time dynamics:**

$$x'_k = \sum_i x_i \frac{f_i(\bar{x})}{\phi(\bar{x})} Q_{ik} \sum_j \frac{2f_i(\bar{x})}{f_i(\bar{x}) + f_j(\bar{x})}$$

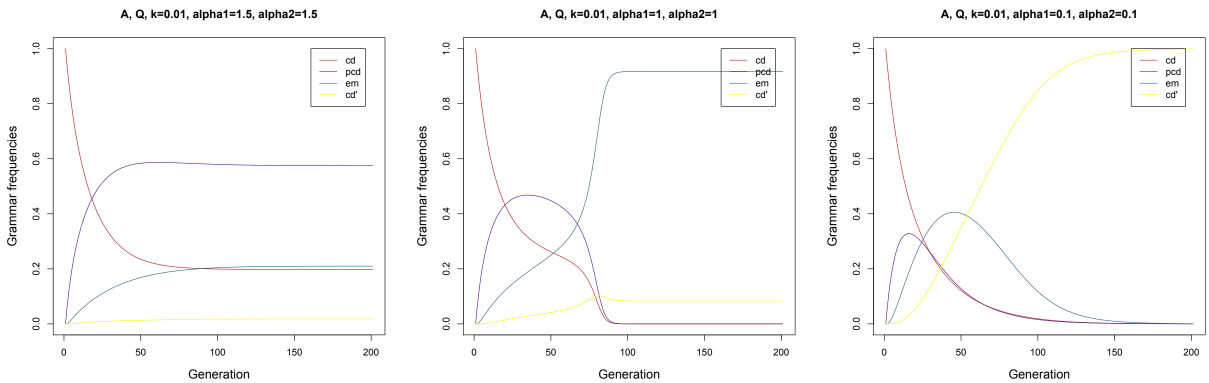
Comparing (43) with the RM dynamics, we can observe that the $\frac{f_i(\bar{x})}{\phi(\bar{x})}$ term has been replaced by term $\frac{f_i(\bar{x})}{\phi(\bar{x})} \sum_j \frac{2f_i(\bar{x})}{f_i(\bar{x})+f_j(\bar{x})}$. One of the consequences is that a grammar with better fitness gets a greater bonus than under the basic RM dynamics. But other than that, the general shape of the evolutionary trajectories remains very similar to what we have already seen:[20]

[19] Here is how we derive (43). In our previous many-parents dynamics in (40), the number of children of i-j pairs was $2x_i x_j$. In the new dynamics, i-speakers will enter into $f_i(\bar{x})/\phi(\bar{x})$ pairs. That means that we will have $2x_i \frac{f_i(\bar{x})}{\phi(\bar{x})} x_j \frac{f_j(\bar{x})}{\phi(\bar{x})}$ offspring from i-j pairs. The dynamics will then look like this:

$$\begin{aligned}
x'_k &= \sum_i \sum_j 2x_i \frac{f_i(\bar{x})}{\phi(\bar{x})} x_j \frac{f_j(\bar{x})}{\phi(\bar{x})} \frac{f_i(\bar{x})}{f_i(\bar{x}) + f_j(\bar{x})} Q_{ik} \\
&= \sum_i 2x_i \frac{f_i(\bar{x})}{\phi(\bar{x})} Q_{ik} \sum_j \frac{x_j f_j(\bar{x})}{\phi(\bar{x})} \sum_j \frac{f_i(\bar{x})}{f_i(\bar{x}) + f_j(\bar{x})} \\
&= \sum_i 2x_i \frac{f_i(\bar{x})}{\phi(\bar{x})} Q_{ik} \frac{\phi(\bar{x})}{\phi(\bar{x})} \sum_j \frac{f_i(\bar{x})}{f_i(\bar{x}) + f_j(\bar{x})} \\
&= \sum_i 2x_i \frac{f_i(\bar{x})}{\phi(\bar{x})} Q_{ik} \sum_j \frac{f_i(\bar{x})}{f_i(\bar{x}) + f_j(\bar{x})}
\end{aligned}$$

[20] We use two α parameters to modify the strength of the fitness effect, separately for the two fitness-dependent terms. α_1 for $\frac{f_i(\bar{x})}{\phi(\bar{x})}$ is designed as for the RM dynamics, see fn. (15). α_2 for $\sum_j \frac{2f_i(\bar{x})}{f_i(\bar{x})+f_j(\bar{x})}$ is defined as for the many-parents dynamics without fitness-dependent pairing, see fn. (18).

(44) **Possible trajectories under the many-parent fitness-ratio fitness-pairing mutator dynamics:**




```
plot_imperfective_share(run_n_generations(x_start, function(x)
    two_parent_f_ratio_f_reproduction_discrete(x,A,Q,
    alpha1=1.5,alpha2=1.5), 200))
plot_imperfective_share(run_n_generations(x_start, function(x)
    two_parent_f_ratio_f_reproduction_discrete(x,A,Q), 200))
plot_imperfective_share(run_n_generations(x_start, function(x)
    two_parent_f_ratio_f_reproduction_discrete(x,A,Q,
    alpha1=0.1,alpha2=0.1), 200))
```

Given this similarity, there is no reason to think that the stochastic version of the new dynamics would behave significantly differently from what we already saw. We can assume that this RM many-parents dynamics produces the same modeling outcomes in (39).

Replacing Deo's single-parent assumption with a more realistic many-parents one has shown that even though the original assumption was not intuitively appealing, it actually did not affect the modeling outcomes. Of course, we could not determine that before analyzing the consequences of adopting a more realistic assumption. Whether purely analytic or through simulations, such checking is an important step for determining the validity of the original simpler model. But the lesson of this section is that not all unrealistic assumptions are harmful. As a corollary, facing a model with a number of unrealistic assumptions, we should not claim that it is a non-starter. Instead, we should check whether the worrisome assumptions actually create trouble.

6.4. **Building an evolutionary dynamics from linguistic assumptions.** In this section, we will illustrate how to derive an evolutionary dynamics from adopted theoretical assumptions about the language change process. In the process, we will see that some sets of evolutionary assumptions that seem reasonable *a priori* will produce undesired predictions when combined with Deo's analysis of communication between speakers of progressive-imperfective grammars encapsulated in the NF ImpGame. In other words, we will demonstrate how modeling can lead to falsification of *a priori* reasonable theories.

One aspect of Deo's model is that switches between grammars occur at fixed rates, given by Q, which do not depend at all on the attractiveness (=fitness) of the target grammars. Moreover, the mutation rates in Deo's Q crucially differ for different pairs of source and target grammars, though at the moment we have no practical protocols that would test how much independent justification the values selected by the analyst may have. So it might be attractive to formulate a different model of the language-change process that would be spared those problems.

Here is how such a model may look like. Offspring agents in the new model have only one linguistic parent, for simplicity. (If we identify agents with time-slices of individuals rather than whole individuals, single parenthood assumption becomes especially appropriate.) Normally, offspring inherit their parent's grammar. But an r share of them, where r is common for all grammars, explores if they can do better. They try out the other strategies, and switch to one of them with likelihood proportional to that strategy's current fitness. In the first variant of the dynamics, such switcher offspring consider all possible grammars. In the second variant, they only consider the neighboring grammars in the cycle — e.g., the child of an *em* speaker would only consider *pcd* and *cd'*, but not *cd*.

The crucial intuitive benefit of the new model — let's call it Constant-Rate Switch Attempt, or CRSA, — is that misacquisition is guided by current comminicative efficiency. This is an appealing assumption whether we take agents to represent individuals or their time-slices. If we choose the former, imagine a child who misacquires a grammar that turns them into a very poor comminicator. It would be useful for such a child to, so to speak, have a second chance, and try to switch to another, more successful grammar. Whether in reality humans can correct such misacquisition mistakes is an empirical question, of course. But given how frequent U-shaped trajectories in language acquisition are, it would be very surprising if the progressive-imperfective part of the grammar allowed all children only one chance to settle on a reasonably successful system.

If, on the other hand, our agents are taken to be time-slices, this means we already presuppose adults' ability to change their progressive-imperfective grammar over the lifetime. In other words, there are definitely "second chances" built into such a model. And where there are second chances, we may reasonably expect agents to be guided by self-interest, i.e. to attempt to maximize their communicative fitness.

Let us now derive the equations for the new CRSA dynamics, starting with the variant where switchers attempt all the grammars. Consider strategy i and its share x_i of the current population. Out of the x_i share, $x_i(1-r)$ speakers will just stay committed to i. As for the other $x_i r$, they will try to switch to other grammars based on their current fitness. A simple law for their choice may be formulated as follows: $x_i r$ speakers represent the whole "switcher pie", and each strategy claims the share of that pie proportional to its fitness. This means that the share of i speakers who will adopt j will be $x_i r \frac{f_j}{f_i + f_j + ...f_n}$, where $f_i + f_j + ...f_n$ is the sum of all fitnesses. The new share x'_i is then the sum of non-switchers $x_i(1 - r)$ and of switchers to i from all strategies:

(45)

$$x'_i = x_i(1 - r) + \sum_j x_j r \frac{f_i}{f_i + f_j + ...f_n}$$

After simplifying the formula using the fact that $\sum_j x_j$ is by definition 1 and that the sum of all fitnesses in the denominator under $\sum_j$ does not depend on j, we get the following form for the dynamics:[21]

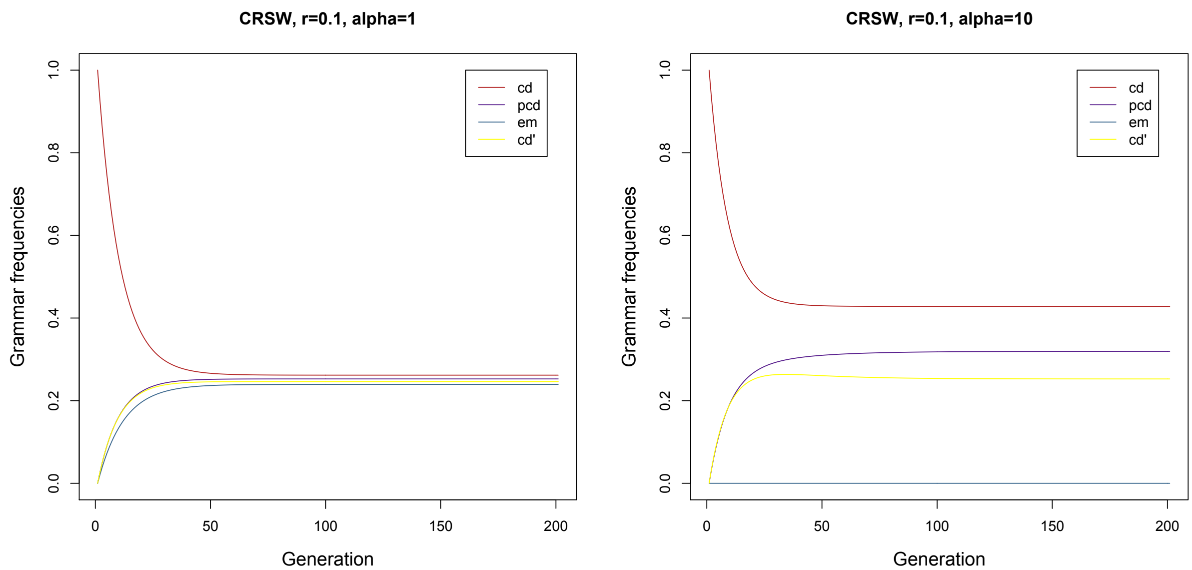
(46) **The CRSA all-grammars discrete-time dynamics:**

$$x'_i = x_i(1 - r) + \frac{r f_i(\bar{x})}{f_i(\bar{x}) + f_j(\bar{x}) + ...f_n(\bar{x})}$$

So what does this model of change predict once we plug into it Deo's A, the payoff matrix of her NF ImpGame? The results are not impressive, as (47) shows. With a low α, we get coexistence of all four strategies. With a high α, the *em* strategy is excluded from the equilibrium, but the other three live happily side by side. This is surely not what we want to derive while modeling the progressive-imperfective cycle!

[21] In practice, we all add parameter α regulating how much effect fitness has. For CRSA, it is defined the same way as for the many-parents dynamics, see fn. (18).

(47) **CRSA all-grammars evolutionary trajectories:**




```
plot_imperfective_share(run_n_generations(x_start,
    function(x) CRSA_all_discrete(x,A,r=0.1, alpha=1), 200))
plot_imperfective_share(run_n_generations(x_start,
    function(x) CRSA_all_discrete(x,A,r=0.1, alpha=10), 400))
```

Why such a disappointing result, and can we do better? Analytically studying the equilibria of the CRSA all-grammars dynamics quickly shows that this failure is no accident.

(48) **The CRSA all-grammars equilibrium:**

$$x_i = x_i(1-r) + \frac{rf_i}{f_i + f_j + ...f_n} \quad \Rightarrow \quad x_i = x_i - x_i r + \frac{rf_i}{f_i + f_j + ...f_n} \quad \Rightarrow$$

$$x_i r = \frac{rf_i}{f_i + f_j + ...f_n} \quad \Rightarrow \quad x_i = \frac{f_i}{f_i + f_j + ...f_n} \quad \Rightarrow$$

$$x_i = \frac{\sum\limits_k x_k A_{ik}}{\sum\limits_k x_k A_{ik} + \sum\limits_k x_k A_{jk} + ... + \sum\limits_k x_k A_{nk}} \quad \Rightarrow$$

$$x_i = \frac{\sum\limits_k x_k A_{ik}}{\sum\limits_k x_k (A_{ik} + A_{jk} + ... + A_{nk})} \quad \Rightarrow$$

$$\Rightarrow \quad x_i = \frac{\sum\limits_k A_{ik}}{\sum\limits_{k,l} A_{lk}}$$

As can be seen in (48), there is a single equilibrium for the CRSA dynamics, and that equilibrium is fully defined by the fitness matrix A. The shares of each grammar i at the equilibrium are simply the sum of i-s fitnesses divided by the sum of all fitnesses in A. Rate r has no effect here whatsoever. Scaling parameter α only shifts the position of the equilibrium, as we replaces the fitnesses from A with their adjusted counterparts.

Let us show that this only equilibrium is the inevitable convergence point for all processes under the CRSA all-grammars dynamics. From (48) we can observe that the switcher term itself does not depend on the current population state $\bar{x}$:

(49)
$$\frac{rf_i}{f_i + f_j + ...f_n} = \frac{r\sum\limits_k A_{ik}}{\sum\limits_{k,l} A_{lk}}$$

Let's call x_i^{eq} the equilibrium value for x_i, namely $\frac{\sum_k A_{ik}}{\sum_{k,l} A_{lk}}$. Consider an arbitrary $x_i = x_i^{eq} + a$. We can then compute the next i-share x_i':

(50) **Approaching the CRSA all-grammars equilibrium:**

$$x_i' = (x_i^{eq} + a)(1 - r) + rx_i^{eq} \quad \Rightarrow$$
$$x_i' = x_i^{eq} + a - rx_i^{eq} - ar + rx_i^{eq} \quad \Rightarrow$$
$$x_i' = x_i - ar$$

In words, the next i share will always get closer to the equilibrium value x_i^{eq} by amount ar, where $a =^{def} x_i - x_i^{eq}$. This shows that there simply cannot be any evolutionary trajectories more interesting than the ones in (47): under the CRSA-all dynamics, all shares always travel towards their equilibrium values with the speed governed by rate r.

What has gone wrong? One conspicuous feature of the trajectories we obtained, shown in (47), is that it is not just *pcd*, the second grammar of the cycle, but also the other grammars whose share start growing immediately, including *cd′* which is not supposed to be available at the start of the cycle. We can modify our dynamics, adding the assumption that speakers of any grammar in the cycle may only shift to the neighboring grammars. As before, for each grammar i there will be a pool of switchers of size rx_i, but unlike before, that pool will be shared between at most three strategies: i's predecessor, i and i's successor.

(51) **The CRSA neighbor-grammars discrete-time dynamics:**

$$x_i' = x_i(1-r) + (x_{i-1} + x_i + x_{i+1}) \frac{r f_i(\bar{x})}{f_{i-1}(\bar{x}) + f_i(\bar{x}) + f_{i+1}(\bar{x})}$$

But this move does not help, as can be readily seen from equilibrium analysis. Note that the term $\frac{f_i(\bar{x})}{f_{i-1}(\bar{x})+f_i(\bar{x})+f_{i+1}(\bar{x})}$ depends only on A, not on the population state $\bar{x}$, just as the corresponding term of the all-grammars dynamics. Denoting that term by K (for constant) for simplicity, we can obtain the following:

(52) **The CRSA neighbor-grammars equilibrium:**

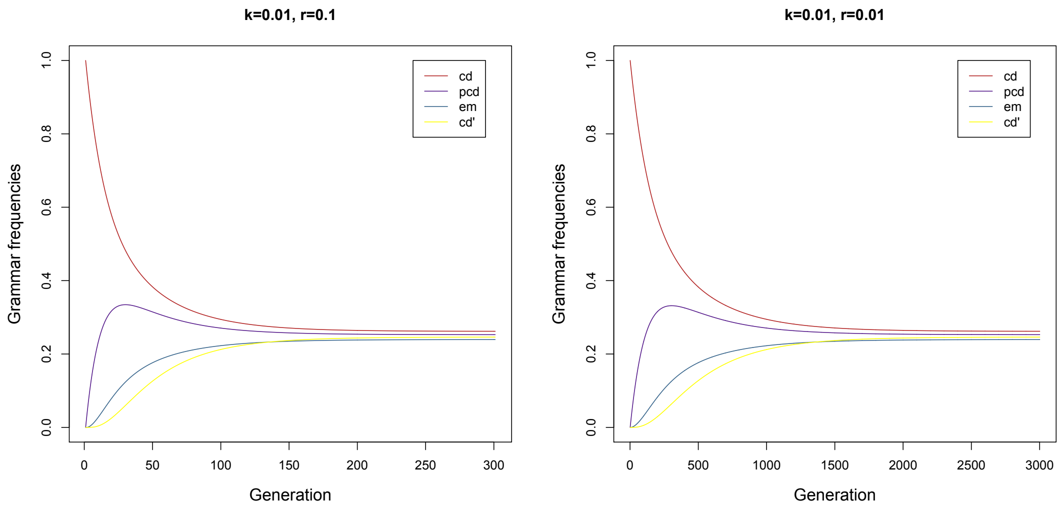
$$x_i = x_i(1-r) + r(x_{i-1} + x_i + x_{i+1})K \quad \Rightarrow$$
$$r x_i = r(x_{i-1} + x_i + x_{i+1})K \quad \Rightarrow \quad x_i - x_i K = x_{i-1}K + x_{i+1}K \quad \Rightarrow$$
$$x_i = \frac{K}{1-K}(x_{i-1} + x_{i+1}) \quad \Rightarrow \quad x_i = \frac{\sum_k A_{ik}}{\sum_k A_{(i-1)k} + A_{(i+1)k}}(x_{i-1} + x_{i+1})$$

Thus we have a very simple system of linear equations describing the equilibrium. Let's refer to $\frac{\sum_k A_{ik}}{\sum_k A_{(i-1)k}+A_{(i+1)k}}$ as K_i, the constant parameter for grammar i provided by the $i-1$, i and $i+1$ rows of the fitness matrix A. Then we have the following in our Evolutionary ImpGame:

(53)
$$x^{eq}_{cd} = K_{cd} x^{eq}_{pcd}$$
$$x^{eq}_{pcd} = K_{pcd}(x^{eq}_{cd} + x^{eq}_{em})$$
$$x^{eq}_{em} = K_{em}(x^{eq}_{pcd} + x^{eq}_{cd'})$$
$$x^{eq}_{cd'} = K_{cd'} x^{eq}_{em}$$
$$x^{eq}_{cd} + x^{eq}_{pcd} + x^{eq}_{em} + x^{eq}_{cd'} = 1$$

The solution for this system is $x^{eq}_{cd} \approx 0.261$, $x^{eq}_{pcd} \approx 0.252$, $x^{eq}_{em} \approx 0.241$, $x^{eq}_{cd'} \approx 0.246$. We can now run a simulation to see how this equilibrium will be approached dynamically (as well as double-check that we found the right equilibrium):

(54) **CRSA neighbor-grammars evolutionary trajectories under Deo's A with $k = 0.01$ under rates $r = 0.1$ and 0.01:**




```
plot_imperfective_share(run_n_generations(x_start,
    function(x) CRSA_neighbor_discrete(x,A,r=0.1), 300))
plot_imperfective_share(run_n_generations(x_start,
    function(x) CRSA_neighbor_discrete(x,A,r=0.01), 3000))
```

As we can see from (54), our additional requirement that only switches between neighboring grammars are allowed did not help much. Though it has created a little boost for *pcd* and a little lag for *em* and *cd′*, this does not change the long-term behavior by much. Discussing the evolutionary behavior of the RM-dynamics models, we noted that they do not fully succeed in capturing the progressive-imperfective cycle. But they were still able to get many details right, such as establishment of at least some strategies as dominant. In comparison, both CRSA dynamics lead to a completely implausible — for the progressive-imperfective cycle — equilibrium with coexistence of all grammars.

At this point, we have a disjunctive result: either the CRSA dynamics are not appropriate for the progressive-imperfective cycle, or Deo's analysis of relative fitness for speakers of different grammars is wrong. But our general analysis of the equilibria for CRSA allows us to generalize easily. No matter what our fitness matrix is, the equations in (53) will ensure that the equilibrium simply involves having all grammars with different frequencies. Therefore the CRSA dynamics are inappropriate for the progressive-imperfective cycle regardless of whether Deo's analysis captured in NF ImpGame is right.

This result illustrates the value of modeling: just looking at our assumptions about the process of change without implementing an explicit evolutionary analysis, we may have never noticed that there was a problem with CRSA![22]

6.5. **BNN-mutator: direct dynamics are inappropriate for the progressive-imperfective cycle with Deo's A.** We finish this section by returning to the BNN-mutator dynamics we introduced in Section (3.2). We will show that BNN-mutator is an obvious non-starter for the progressive-imperfective cycle when coupled with Deo's NF ImpGame. Recall how that dynamics is defined:

(13) **Discrete-time BNN-mutator dynamics**

$$x'_i = \sum_k \frac{x_k + \alpha[f_k(\bar{x}) - \phi(\bar{x})]_+}{1 + \sum_j \alpha[f_j(\bar{x}) - \phi(\bar{x})]_+} Q_{ki}$$

A crucial feature of BNN-mutator is that at the fitness step an increase may occur even to strategies that are not present in the population. In the RM, the fitness step is defined by $x_j \frac{f_j(\bar{b})}{\phi(\bar{x})}$, where the new fitness-adjusted x'_j is related to x_j through a coefficient. Therefore for $x_j = 0$, better fitness cannot increase j's share. It is only mutation that can. But that is not so in BNN: even if x_k is 0, the share of x_k may increase at the fitness step. Dynamics like the replicator dynamics are called *imitative*: only those strategies may expand which are already present in the population, as if only the observed strategies may be adopted by switching agents. The BNN dynamics, in contrast to that, is *direct*: the changes in the current shares

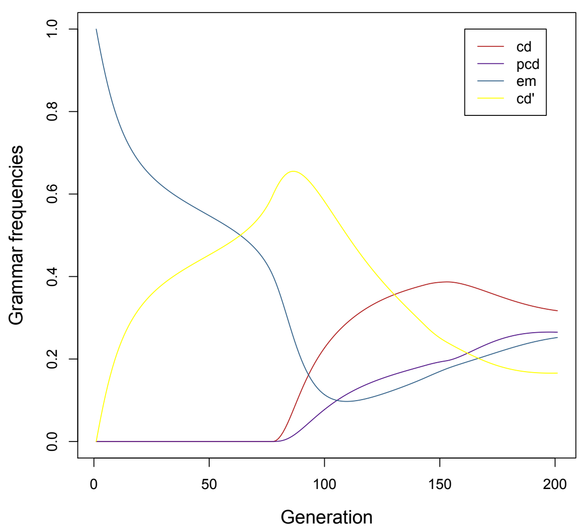
[22] Our analysis also allows us to speculate about where a CRSA-like model *could* be useful. It is a very common phenomenon in language use to have structured variation between several forms that are in some sense linguistic equivalents of each other. Such alternative variants form what sociolinguists call a *variable*. An example of a variable is pronunciations of [t] or [θ] in words like *thin*. (See [Tagliamonte, 2012] for an introduction into modern variationist sociolinguistics.) The fitness of a given construction forming the variable may be taken to be its average appropriateness given the structure of the speaker's social interactions: for example, with some conversational counterparts [t] would be more appropriate, while with others it would be met with disapproval. Fitnesses will thus depend on another parameter in addition to the frequency of the variants themselves, namely on the social situation of the variants' bearer.

With changes in that external parameter, fitnesses of the variants, and therefore the CRSA equilibrium, will change. A priori we cannot tell whether the trajectories of moving towards the new equilibrium (such as, for example, when the bearer of our variants changes their social situation by moving into another area or getting a different job) would look like CRSA trajectories. But at least CRSA would not be an *a priori* inappropriate choice for modeling this. So one and the same mathematical model of language change may be appropriate for one type of change, but not another.

are defined on the basis of all the current fitnesses, even those of strategies that are not currently used in the current population. (This is called *direct* because agents in this setup have direct access to all strategies, as opposed to having access only to those which they can observe.)

Now, as we also have the mutation step in our dynamics that shifts agents around, the effect of this distinction is a bit less than in mutator-less cases: mutation also may introduce agents with strategies that did not already exist in the population. But in Deo's NF ImpGame, there is no out-mutation from the strategies *em* and *cd′* into the first two strategies *cd* and *pcd*. This is where the crucial feature of BNN-mutator makes things go conspicuously wrong:

(55) **BNN-mutator dynamics with Deo's A with $k = 0.01$ and Q, starting from the all-*em* state:**




```
plot_imperfective_share(run_n_generations(c(0,0,1,0),
    function(x) BNN_mutator_discrete(x,A,Q),200) )
```

We see in (55) that after some competition between *em* and *cd′*, the initial grammar *cd* suddenly makes a comeback. This should not be possible in the progressive-imperfective cycle: after X is already fading out from the language, we do not want to see the grammar that only uses X for both the progressive and the imperfective to come back and even overtake the all-Y grammar *cd′*. Yet this is what the BNN-mutator dynamics predicts. This is so because of the structure of Deo's A from the NF ImpGame:

(6) **NF ImpGame A:**

Strategies	cd	pcd	em	cd'
cd	0.9	0.9	0.7	0.9
pcd	$0.9 - \frac{k}{2}$	$0.95 - \frac{k}{2}$	$0.75 - \frac{k}{2}$	$0.7 - \frac{k}{2}$
em	$0.7 - \frac{k}{2}$	$0.75 - \frac{k}{2}$	$1 - \frac{k}{2}$	$0.7 - \frac{k}{2}$
cd'	0.9	0.7	0.7	0.9

Note that *cd* plays equally well or better against every strategy than *cd*′. Thus according to A alone, *cd* is actually a better or equally good choice. As BNN-mutator has a direct, not imitative, fitness step, *cd*'s good fitness allows it to be reintroduced into the population.

An indirect conclusion we can draw is that Deo's choice of Q where there is no mutations back from *em* and *cd*′ was crucial for deriving results close to the desired predictions. Not allowing *cd* and *pcd* arise because of their fitness was one important thing; not allowing back-mutations into those was another. If mutations allowed some *cd* speakers to arise in an *em*-*cd*′ population, fitness-wise those speakers would be OK.

6.6. **Summary: different dynamics in the Ev ImpGame.** In this section, we have looked more closely at the linguistic interpretation of the replicator-mutator dynamics in the Evolutionary Imperfective Game. We discussed two possibilities: Deo's own interpretation where agents correspond to people, and our alternative interpretation where agents are people's time-slices. Though both interpretations include some not intuitively appealing assumptions, we argued that one should not discard the RM dynamics just because of that without checking how harmful those are. We illustrated that with the assumption of single-parenthood, required by Deo's interpretation of the RM. It turned out that changing a one-parent setup to a two-parent setup did not affect the predictions of our modeling by much.

After that, we examined two different evolutionary dynamics. Starting from a set of seemingly plausible (though by no means obviously correct) assumptions about language change, we defined the CRSA dynamics. It quickly became evident that despite it being grounded in seemingly reasonable assumptions, CRSA performs terribly in the progressive-imperfective cycle. We then considered the BNN-mutator dynamics, and showed that it also is a non-starter in this case, as it allows and in fact favors reintroduction of the earlier grammars from the cycle. Though not built from a preformulated set of assumptions as CRSA, the assumptions of the BNN-mutator dynamics at the first glance looked not less plausible than those of the RM dynamics, Section 3.2.

There are three points we can draw from this. First, not all intuitively terrible assumptions are actually terrible for modeling purposes. Second, not all intuitively plausible assumptions will work well. Finally, examining one's assumptions is important. For example, it could have turned out that switching to the many-parents setting would have harmed our predictions. If we did not spell out the implicit assumptions of our dynamics and check this one, we could have missed a factor that makes our modeling results irrelevant.

7. Evolution of utterances instead of grammars?

In Deo's framework, evolution applies to grammars. Whether we take them to be immutable grammars acquired in childhood (as Deo herself does), or current grammars that characterize speakers' behavior during a limited period of time (as we suggested in Section 6.2), it is grammars that compete in this framework.

In a different evolutionary framework outlined by [Croft, 2000], it is (individual aspects of) utterances that compete with each other and are thus subject to evolution. Under this approach, language change is analyzed not (only) in terms of changes in the underlying speaker grammars, but in terms of changing distributions of utterances.

Though the number of actual evolutionary analyses of specific linguistic changes is currently very small for both frameworks, one might be tempted to determine once and for all which of them is "the right one". In this section, I outline what an utterance-based analysis of the progressive-imperfective cycle would look like, and conclude that the choice between the grammar-based and utterance-based options should be a matter of convenience for the particular analysis to be undertaken, not a matter of which one is "right" or "wrong".

7.1. **Sketching an utterance-based model of the progressive-imperfective cycle.** To see what objects will be undergoing an utterance-based evolutionary process, we need to think about what changes in the actual set of utterances happen between the four stages of the progressive-imperfective cycle. If we adopt Deo's formal interpretation of the four stages, then, for example, at the initial stage we will only see linguistic form X. That form will be able to occur in both contexts C_{phen} and C_{struc} and with both intended meanings **phen** and **struc**. In contrast, at the categorical-progressive stage (Deo's *em*), we only observe X used with meaning **struc**, and Y with **phen**, but again regardless of context. Thus the objects of reproduction will be triples like $\langle X, C_{struc}, \mathbf{phen}\rangle$. We can abbreviate $\langle X, C_{struc}, \mathbf{phen}\rangle$ as Xsp, and do accordingly with all other triples. Here are the triples that we expect to (mostly) see at the different stages of the cycle:

(56) (i) *cd*: Xss, Xsp, Xpp, Xps
(ii) *pcd*: Xss, Ysp, Xpp, Xps
(iii) *em*: Xss, Ysp, Ypp, Xps
(iv) *cd′*: Yss, Ysp, Ypp, Yps

Note that the distribution of contexts and intended meanings is not meant to be subject to evolution. We should model it as constant, or nearly constant: we do not assume that in earlier times, people talked more about progressive meanings, etc.

In accordance with Deo's model, we can take the frequency of C_{phen} and C_{struc} to be equal, and the frequency of context-mismatching messages to be 0.1. Then (57) shows how the population of utterances at some point in the evolutionary process might look like. It is easy to see that the distribution in (57) is very close to the idealized *pcd* stage.

(57)

Xss	Yss	Xsp	Ysp	Xpp	Ypp	Xps	Yps
0.39	0.06	0.002	0.048	0.41	0.04	0.049	0.001

To formulate an evolutionary analysis, we will need to state how such population states develop. In particular, we will need to assign them fitnesses that represent usefulness to the speakers (e.g., communicative success), and decide how those fitnesses will guide the evolutionary process. To assess communicative success, we should consider the distribution of *observable* tuples corresponding to (57):

(58)

Xs	Ys	Xp	Yp
0.392	0.108	0.459	0.041

Communicative success will be based on how well the distribution in (57) can be recovered from (58); that is, how many misclassifications a language user would make when going from the latter to the former. Obviously there are many rules (i.e. hearer grammars) that allow one to guess what (57) is given (58). We need to formulate a theory of how the agent would learn a grammar upon observing (58). That learned grammar may then be used to generate a new distribution of the eight forms. That distribution will be the offspring of (57).

Formulating the grammar-learning procedure for our agents, we need to make a major theoretical choice about the grammar space available to the learner. If we attempted to translate Deo's analysis of the progressive-imperfective cycle into the utterance-based framework with minimal modifications, we could say that our agents only have four grammars to choose from: Deo's *cd*, *pcd*, *em* and *cd′*. We will see below that this would not be a viable choice after all, but for now, let's just note that learning in this setting amounts to choosing one grammar out of four, and that there are reasonable ways to do that, such as maximal likelihood estimation (i.e. finding the grammar which is most likely to have generated the observations). If our learning procedure is reasonable at all, it should not be hard to choose *pcd* upon having seen

(57). Intuitively, this would result from the following: Xs and Xp occur much more frequently than in 50% of cases in (57), thus making *em* and cd' grammars very improbable; furthermore, Ys occurs twice as much as Yp, which is more likely under *pcd* than *cd*.

Alternatively, we can side with, among others, the sociolinguistic tradition where grammars are inherently probabilistic. A probabilistic grammar can look as in (59). Having a defined grammar space, we can also rely on maximum likelihood estimates for deriving the most likely probabilistic grammar from the observed distribution (58).

(59)
$$\begin{array}{c} \mathrm{X} \leftarrow^{0.89} \mathrm{ss}\ {}^{0.11} \rightarrow \mathrm{Y} \\ \mathrm{X} \leftarrow^{0.05} \mathrm{sp}\ {}^{0.95} \rightarrow \mathrm{Y} \\ \mathrm{X} \leftarrow^{0.93} \mathrm{pp}\ {}^{0.07} \rightarrow \mathrm{Y} \\ \mathrm{X} \leftarrow^{0.97} \mathrm{ps}\ {}^{0.03} \rightarrow \mathrm{Y} \end{array}$$

And now we come to the hard part, which is also the interesting part. If learning a grammar and generating the next utterance distribution is all there is, we cannot get any directed development that is needed for the progressive-imperfective cycle. Even if we add some random noise to the production stage, this noise would sometimes give us more Ys, but other times more Xs. In order to derive movement from an all-X utterance pool towards an all-Y one, we need to introduce a factor that would favor Ys over Xs: mutation of utterances.

Let's assume that in production, each agent generates output according to their grammar. Then a mutation stage occurs: some of the agent's utterances end up having a different message instead. So even if the agent's grammar says that they should only generate Xs, there may still occur some Ys in the output because of mutation. Let's also assume that the $X \rightarrow Y$ mutation rate is greater than the $Y \rightarrow X$ mutation rate.

What will the evolutionary process then look like in the 4-grammar space? Consider for simplicity the all-X starting distribution. Learning a grammar from it, a rational agent should choose the *cd* grammar that only has signal X. Then in production, they will also produce only Xs. Some of those will be turned into Ys due to mutation. But unless the mutations are very numerous, at the next round our agent would again learn the all-X *cd* grammar: after all, the Y deviations from it are negligible. So if we work with the 4-grammar space, then at each round we will be thrown back into necessarily learning the *cd* grammar, unless mutations are so strong that they can create a very significant share of Ys in one round. The setting with only a few grammars in the grammar space is thus not very convenient for utterance-based evolutionary modeling. A small grammar space is not fine-grained enough to track the small changes in the frequencies of utterances.

On the other hand, in the wider grammar space of probabilistic grammars, we will expect to see the grammars gradually shifting from all-X to all-Y. The increase of Ys due to mutation will be translated into a slightly higher frequency for Y in the grammar, and thus in the next round, there will be even more Ys.

But a simple argument shows that there will be non-trivial challenges in making this utterance-based setup work. First, if $X \to Y$ mutations occur equiprobably for **phen** and **struc** contexts, we will not derive that Y always becomes a progressive form. Even if categorical signaling emerges, it would be equally likely to employ Y for the imperfective as for the progressive, which is wrong. On the other hand, allowing $X \to Y$ to occur in **phen**, we would push the system to reach the categorical stage *em* where Y is the obligatory progressive marker. But then the process will be stuck there, as there would be no incoming evidence for Yss and Yps, which should overtake Xss and Xps at the final stage of the cycle. So just allowing $X \to Y$ in **phen** is not an option either. Finally, if we allow $X \to Y$ to happen in both **phen** and **struc**, but at a higher rate in the former case, we would eventually derive the final *cd′* stage, but there would be no *em* stage: the probabilities for using Y in **struc** contexts would start growing from the very start, and not only after we reach the categorical system *em*. Therefore some non-trivial modeling assumptions will be needed in order to derive the progressive-imperfective cycle in the utterance-based setting.

7.2. **Comparing the utterance-based and grammar-based models of the progressive-imperfective cycle.** Both the grammar-based and the utterance-based models crucially need mutation of some sort in order to derive the progression of the progressive-imperfective cycle through different stages. For the grammar-based models, mutation changes grammars. In the utterance-based setup, it changes utterances. But in both models, the reasons for such changes are not analyzed further. It is just taken as given that there are forces that either make agents shift grammars, or new utterance types get created. It is only the effect of those unanalyzed forces onto the progressive-imperfective system as a whole that both models analyze.

But with regard to the grammar space, the two models are on different levels of abstraction. Out of many possible combinations, [Deo, 2015] selects without argument four grammars *cd*, *pcd*, *em* and *cd′*, and analyzes how they would interact in a population. Those four grammars to large extent match what we know about the actual examples of the progressive-imperfective cycle. Thus by adopting precisely those grammars as the only possibilities, we spare ourselves the work of deriving the fact that only those would emerge.

Such work would not be trivial, however. For example, why it is the progressive forms and not imperfective ones that spread during the cycle? [Deo, 2015, Sec. 4.3] explicitly discusses this problem, and puts forward a particular hypothesis aimed to solve it. But in her evolutionary modeling, she accepts the observed direction of change as given. In the utterance-based framework, however, we cannot easily dismiss this problem. As we discussed above, it makes little sense to combine the utterance-based framework with a small grammar space. But in a larger grammar space, we would need to derive not just the sequence of four grammars *cd*, *pcd*, *em* and cd', but also the fact that those grammars, and not many possible others, are favored by the evolutionary process. As we have seen in the previous subsection, deriving this would require adopting non-trivial assumptions.

Which of the two approaches is better? That depends. In the case at hand, the grammar-based approach takes more things for granted, incorporating our expert knowledge about the progressive-imperfective cycle into the choice of four grammars. We can then concentrate on deriving the sequence of those. The utterance-based approach, in contrast, has to come up with plausible restrictions that would derive something like Deo's four grammars from more basic assumptions. If successful, such approach would be more explanatory, as we would derive the global properties of the cycle from more elementary facts. But more work would be needed for that, as we would have to work with objects of a lower level of abstraction.

Is it bad that the grammar-based approach takes more things for granted? Not necessarily: as long as we only take for granted well-established things, we are all right.[23] An analogy may be useful here. When analyzing the physics of car movement, we can go low-level and derive the movements of each car from the forces in action in its internal parts: the force of combustion in its engine, how it is transferred to the wheels, how the wheels experience friction, and so forth. In the end, we will derive a cumulative force working on the car as a whole from those elementary components. But we can also take for granted that all mentioned forces together conspire to produce a given cumulative force, and only work with that cumulative force. Depending on what exactly we want to study, either choice may be more appropriate. Choosing between the grammar-based and the utterance-based setup

[23] Actually, we may sometimes be all right even after taking for granted things that were not certain. As an example, [Deo, 2015] assumes that in the second stage of the cycle, the new optional progressive marker Y is only used to signal **phen** in contexts favoring **struc**. While this is a plausible assumption, we do not actually know whether it is how speakers actually behave. But in order to proceed with the analysis, we need to make *some* assumption about how speakers at the second stage behave. Moreover, without the relevant empirical data, a more low-level theory would be just as much at a loss as the theory that takes an unproven assumption for granted. In this particular case, even if we avoid making the *pcd* assumption while working in the utterance-based framework, without data we would not be able to tell if our predictions are on the right track. Our trajectories will either pass through *pcd* or not, but if we do not know which of the two options is correct, we are no better off than in Deo's model where it is assumed that the second stage works like *pcd*.

may be viewed similarly: we should choose one not because it is "right", but because it provides an appropriate level of abstraction for our analytic goals.

Does it change the matter if it turns out that one of the two approaches is indeed crucially more accurate than the other for modeling language change? Not really. For instance, we know that Newtonian mechanics is strictly speaking wrong. But for many phenomena, it can be used with perfect success. Its value is not in its being absolutely correct, but rather in the fact that it allows us to derive predictions that withstand empirical testing well as long as we know the limits of its applicability. So even if one of the two approaches to evolutionary modeling of language change turns out to be strictly more correct than the other, this would not automatically mean that the other one should never be used.

The main practical danger of working at a higher level is taking for granted something which is not yet known to be true, which turns to be false, and moreover affected the predictions from modeling. As a potential example, consider how [Deo, 2015] analyzes the second stage of the cycle, where the innovative progressive marker Y is still optional. Deo constructs strategy *pcd* to represent this stage where Y is only used in context C_{struc} to signal **phen**. We do not really know whether this is how speakers use innovative progressive forms. So we have three possibilities. First, Deo's choice may match the actual data (a good case). Second, the optional-progressive stage of the cycle may look differently from *pcd*, and under a more correct formalization of that stage, our evolutionary predictions would be different from the original ones with *pcd* (a bad case). Third, it can be that there is no actual *pcd* stage, but the real optional-progressive stage would behave the same in modeling as *pcd* does (another good case). The weakness of higher-level approaches is the opposite side of their strength: they allow us to encapsulate in the model many assumptions about how language change works for our linguistic phenomenon of interest, but if those built-in assumptions turn out to be wrong, then we get worse predictions. Note that any theory, including utterance-based ones, would need a lot of assumptions in order to get off the ground. The only difference between higher-level and lower-level theories is that the latter are forced to explicitly model more details of the process, thus requiring us to spell out more assumptions.

8. Conclusion

We started with the evolutionary model for the progressive-imperfective diachronic cycle by [Deo, 2015] and saw how that model behaved under a wide range of parameter settings, Sections 4 and 5. We discussed how other models, based on different evolutionary dynamics, may look like, Section 6. We also sketched how a completely different evolutionary framework by [Croft, 2000], where utterances rather than grammars compete, could analyze the progressive-imperfective cycle, Section 7.

In the course of studying Deo's Imperfective Game, we illustrated a number of practical techniques that may be applied to evolutionary modeling of language change in general. Tools well-known in other areas of evolutionary analysis, such as fitness landscape, forces between strategies, and mathematical equilibrium analysis, can be hopefully of use for modeling linguistic phenomena. Over the course of the paper, we introduced several different modeling setups, and showed how to construct new evolutionary dynamics based on the analyst's favorite assumptions about the process of change. The goal here was to demystify the choice between evolutionary models and the process of building new ones.

Here, we sum up the main insights from our study. First we go over the theoretical points concerning semantic change in general, and then turn to the Imperfective Game and the progressive-imperfective cycle.[24]

General conclusions

- There is not a single best choice of method for evolutionary modeling semantic change, or language change in general. There exist grammar-based vs. utterance-based methods; there exist many different evolutionary dynamics; evolution may be analyzed deterministically or stochastically; parameter values for a given modeling setup may greatly affect the resulting change trajectories.

- Different choices of modeling methods imply different theoretical assumptions about how language change works. This need not prevent one from trying out a given method; however, once our method derives specific results, we need to address the assumptions that it is based on, to check how plausible they are. In a good case, modeling may even provide arguments for some theories of language change against others.

- There is no need to deem either grammar-based or utterance-based evolutionary modeling as the *only* right approach. Any model, in any science, is simpler than the reality and aims to zero in onto some specific aspects of the world. Utterance-based evolutionary approaches require that more details are explicitly modeled. Grammar-based approaches have easier time encapsulating prior expert knowledge about how language works.

[24]Many of the general theoretical points are well-known outside of linguistics. For example, genetic drift would be one of the first topics in Population Genetics 101. I include such points in the general summary not because they are new or surprising, but because they have not been to my knowledge applied in the literature on semantic change.

- Some seemingly innocent assumptions about language change lead to disastrous predictions (Section 6.4). The other way round, some seemingly outrageous assumptions in fact are not particularly problematic, as they do not affect the predictions greatly (Section 6.3).

- Evolutionary dynamics may include components sensitive to functional considerations and components blindly imposing shifts (between grammars or forms). In line with the biological tradition, the former may be conceptualized as *fitness*, and the latter, as *mutation*.

  But unlike in biology, we need not assume a constant across-analyses interpretation for fitness and mutation. For example, when we analyze loss of morphological cases triggered by reduction of case endings, the phonological loss may be treated as mutation: a blind force turning some linguistic systems into others. But when we analyze phonological reduction in its own right, we better regard the reduction forces as fitness forces, associated with reduction of effort. Thus in a specific analysis, we encode in the mutation component those forces that we do not want to analyze further, and in the fitness component, the functional considerations that we think should affect our phenomenon of interest directly.[25]

- In modeling cyclic patterns, it is natural to use fitness for maintaining stable stages of the cycle, and mutation for moving between the stages (Section 4.3).

- Working with finite populations as opposed to infinite ones (Section 5) is not only methodologically more appealing, but also leads to better empirical predictions. In a finite population, evolution is by definition stochastic, just as language change is well known to be.

  In particular, if the evolutionary process has several equilibria corresponding to different stages of some cycle, stochasticity may allow, under the right conditions, the process to cross over from one equilibrium to the next. Stochasticity is thus crucial for deriving sequences of stages each of which is functionally good as it is.

[25]If we were to build the model of a changing language as a whole, presumably there would be no re-classification between fitness and mutation. But building such a gigantic model is too complex an enterprise to undertake, at least as of this stage of research.

- When intermediate one-grammar stages are not equilibria or are equilibria with a very small basin of attraction, smaller population sizes may help nevertheless reach them. This is due to the fact that genetic drift, which favors uniform populations, becomes stronger in smaller populations.

Conclusions and open questions specific to the Imperfective Game and the progressive-imperfective cycle

- We examined the behavior of the Evolutionary Imperfective Game under the RM, fitness-ration many-parent mutator, CRSA and BNN-mutator dynamics. Applied on top of Deo's analysis of the interactions of speakers with different progressive-imperfective grammars (which is encoded in the NF ImpGame A), the first two produce reasonable predictions, though not the ones *a priori* expected. The last two just do not work at all.

- Why do the RM and many-parent mutator dynamics *both* work reasonably well? They are associated with quite different sets of assumptions about language change. Can something deep follow from the facts regarding which dynamics produce decent results in the Ev ImpGame?

  Deo's original interpretation of the RM dynamics implies that each agent corresponds to one person's lifetime, that people only have a single "linguistic parent", and people with a communicatively better imperfective-progressive segment of grammar have significantly more "linguistic children". Our alternative interpretation is that each agent is a person's time slice. The next slice has a single "parent": that very person's previous time-slice. More communicatively successful persons persuade less successful communicators to switch to their grammar. In our interpretation for the many-parents dynamics, each agent has two or more linguistic parents, and chooses its grammar based on the relative communicative efficiency of the parents' grammars.

  The multiplicity of reasonably working models and interpretations means that it is difficult to derive from evolutionary modeling *non-ambiguous* conclusions about the actual mechanisms of language change. External evidence seems to be needed in order to decide between the different models.

- Deterministic, infinite-population dynamics are a poor choice for the Ev ImpGame. Switching to the finite-population setting creates stochasticity that makes the evolutionary trajectories look much more similar to what we expect given the research within historical linguistics.

  However, stochasticity only becomes significant at small population sizes (e.g., of 100 individuals). It is currently unclear how to interpret this, but there is one plausible line of investigation. The relevant population-size range may reasonably characterize real-life small communities of frequently interacting speakers. It is an open question how the evolutionary process of the Ev ImpGame will behave in a network of loosely connected small communities. If reasonable aggregate results arise in such a setting, the small number of individuals needed for adding enough randomness will not be a problem.

- The perspective of an utterance-based evolutionary framework in the style of [Croft, 2000] highlights further challenges for modeling the progressive-imperfective cycle. Deo's grammar-based evolutionary model abstracts away from how individual innovative utterances appear where the old form X is replaced by the innovated Y. But when we try to account for the spread of such utterances explicitly, it becomes clear that we need additional substantive assumptions that would force innovations arise with different frequencies in different contexts. Much future research is needed on this front: utterance-based models should be able to derive the observed sequence of grammars rather than assume it as given, and our initial examination shows that this is not an easy task.

- Crucially, even the grammar-based dynamics that we called reasonably working, namely the RM and many-parents dynamics, do not derive exactly the expected sequence $cd \to pcd \to em \to cd'$ in the Ev ImpGame. There are three options that emerged in our analyses in Section 5: (1) no *pcd* stage, but progression into *em* and then cd'; (2) good *pcd* and *em* stages, but no progression into cd'; (3) good progression $cd \to pcd \to em \to cd'$ in many runs, but also skipping of stages in some simulations, and back-shifts in yet others.

  Though those modeling outcomes are not what we expected for the progressive-imperfective cycle, we currently do not have diachronic data fine-grained enough to say with certainty that those predictions are wrong, especially for outcomes (2) and (3). More empirical research is needed to establish the actual evolutionary trajectories of real-life progressive-imperfective cycles.

References

[Baxter et al., 2009] Baxter, G. J., Blythe, R. A., Croft, W., and McKane, A. J. (2009). Modeling language change: An evaluation of Trudgill's theory of the emergence of New Zealand English. *Language Variation and Change*, 21:257–296.

[Blythe and Croft, 2012] Blythe, R. A. and Croft, W. (2012). S-curves and the mechanisms of propagation in language change. *Language*, 88(2):269–304.

[Bybee et al., 1994] Bybee, J. L., Perkins, R., and Pagliuca, W. (1994). *The evolution of grammar: Tense, aspect and modality in the languages of the world.* University of Chicago Press, Chicago.

[Croft, 2000] Croft, W. (2000). *Explaining language change: an evolutionary approach.* Longman, Harlow, Essex.

[Deo, 2015] Deo, A. (2015). The semantic and pragmatic underpinnings of grammaticalization paths: The progressive to imperfective shift. *Semantics and Pragmatics.*

[Eckardt, 2006] Eckardt, R. (2006). *Meaning Change in Grammaticalization. An Enquiry into Semantic Reanalysis.* Oxford University Press, Oxford.

[Ellegård, 1953] Ellegård, A. (1953). *The auxiliary* do*: the establishment and regulation of its use in English.* Almqvist and Wiksell, Stockholm.

[Ernout, 1914] Ernout, A. (1914). *Morphologie historique du Latin.* C. Klincksieck, Paris.

[Fleischman, 1982] Fleischman, S. (1982). *The future in thought and language. Diachronic Evidence from Romance*, volume 36 of *Cambridge Studies in Linguistics.* Cambridge University Press.

[Jespersen, 1917] Jespersen, O. (1917). *Negation in English and other languages.* Høst, Copenhagen.

[Labov, 1994] Labov, W. (1994). *Principles of linguistic change. Volume I: Internal factors.* Wiley-Blackwell.

[Lightfoot, 1991] Lightfoot, D. (1991). *How to set parameters: Arguments from language change.* MIT Press, Cambridge, MA.

[Nash, 1951] Nash, J. (1951). Non-cooperative games. *The Annals of Mathematics, Second Series*, 54(2):286–295.

[Raumolin-Brunberg, 2009] Raumolin-Brunberg, H. (2009). Lifespan changes in the language of three early modern gentlemen. In Nurmi, A., Nevala, M., and Palander-Collin, M., editors, *The Language of Daily Life in England (1450-1800)*, pages 165–196. Benjamins.

[Seoane, 2012] Seoane, E. (2012). Early Modern English: Syntax. In Bergs, A. and Brinton, L. J., editors, *English Historical Linguistics: An International Handbook*, volume 1, pages 621–637. De Gruyter Mouton.

[Tagliamonte, 2012] Tagliamonte, S. (2012). *Variationist Sociolinguistics.* Wiley-Blackwell.

[Tellier, 1962] Tellier, A. (1962). *Les verbes perfecto-présents et les auxiliaires de mode en anglais ancien: (VIII^e S. - XVI^e S.).* C. Klincksieck, Paris.

Appendix A. Explaining the structure of the R code for generating diagrams

Here, we explain the parts that make up the R command that creates the diagram in (9). The R files in the supplementary materials contain more information about how different functions work and what their arguments mean.

```
plot_imperfective_share(run_n_generations(c(1,0,0,0),
    function(x) replicator_mutator_discrete(x,A,Q), 150) )
```

`c(1,0,0,0)` defines the initial population state. In this case, the population has 4 strategies; the share of the first one, x_1, is 100%, while the shares of the other three are 0.

`replicator_mutator_discrete` is the name of the replicator-mutator dynamics function, just as defined in (8). It takes a population state, A and Q as arguments, and returns the next population state.

Function `run_n_generations` runs the dynamics function a high number of times to obtain a long-term evolutionary trajectory. Its first argument is the starting state, in our case `c(1,0,0,0)`. Its second argument is a wrapper function that only has one argument, the current population state, and returns the next population state. We define this needed function using a λ expression: `function(x)` plays the same role as λx. Note that we are feeding our predefined matrices A and Q to `replicator_mutator_discrete`. Using other matrices will lead to different results. Finally, the last argument to `run_n_generations`, in this case 150, is the number of generations that we want to track.

The output of `run_n_generations` is a data structure listing each population state that was generated. We can look it up directly in R if we save it into a variable (in this case, named `deo`), and then simply type the variable's name into R's command prompt:

```
deo = run_n_generations(c(1,0,0,0),
    function(x) replicator_mutator_discrete(x,A,Q), 150)
deo
```

This will produce many lines of output, starting with what is shown below. Each line here lists a generation:

```
             [,1]         [,2]        [,3]         [,4]
[1,] 1.000000e+00 0.000000e+00 0.000000000 0.000000e+00
[2,] 9.400000e-01 6.000000e-02 0.000000000 0.000000e+00
[3,] 8.849153e-01 1.108935e-01 0.004191225 0.000000e+00
[4,] 8.347304e-01 1.542201e-01 0.010951251 9.821964e-05
[5,] 7.891521e-01 1.912722e-01 0.019221913 3.538244e-04
...
```

Finally, function `plot_imperfective_share` creates a plot that represents the information generated by `run_n_generations`. If we saved the output of `run_n_generations` in variable `deo`, we can apply `plot_imperfective_share` directly to `deo` to display it:

```
plot_imperfective_share(deo)
```