A Comparison of Lexicographic and Linear Numeric Optimization Using Violation Difference Ratios*

Bruce Tesar
Rutgers University
12/1/07

1 Introduction

Optimality Theory (henceforth OT) (Prince and Smolensky 1993/2004) is based upon lexicographic optimization. It differs in this respect from Harmonic Grammar (henceforth HG) (Legendre et al. 1990a, Legendre et al. 1990b), which is based upon linear numeric optimization. Differences between the two have been discussed in several places, including (Legendre et al. 2006, Pater et al. 2007a, Prince and Smolensky 1993/2004).[1]

Patterns which are achievable in HG but not in OT can be called **cumulative interactions[2]**. Extensive discussion and exemplification of cumulative interactions is provided in (Pater et al. 2007a) (henceforth PBP). An important point of PBP is that HG cannot realize every possible pattern: only some linguistic patterns which are not achievable in OT can be achieved via cumulative interactions. In part of their paper, PBP examine some undesirable cumulative interactions, in particular ones that occur when linear numeric optimization is used in the context of "global" optimization models, in which the optimal candidate(s) in a competition is decided via a single optimization in which all candidates are simultaneously competing. PBP ultimately propose a variant of Harmonic Serialism (Prince and Smolensky 1993/2004), called Local Harmonic Serialism, to prevent the undesired interactions. Local Harmonic Serialism draws in part on recent work by McCarthy (McCarthy 2006b, McCarthy 2007). Harmonic Serialism uses "local" optimization, in which the optimal candidate is determined via a series of optimizations, each of which consists only of candidates that differ in a specific localized property (see PBP for further discussion).

In this paper, I focus exclusively on "global" optimization models, providing further investigation into the similarities and differences between OT and HG in that context. The relation is viewed with respect to violation differences: the difference (as a quantity) in the number of violations assessed to two candidates in a comparison. When two constraints conflict on a candidate comparison, we can examine the violation difference ratio of those constraints: the relative size of the differences in the number of violations for the constraints. The relation between violation difference ratios for different candidate comparisons can predict potential for cumulative interactions.

The major points of the present work are listed in (1).

---

* The author would like to acknowledge beneficial interactions with Jane Grimshaw, Alan Prince, and the Rutgers Optimality Reading Group, and beneficial comments from Joe Pater on a prior draft of this paper. All mistakes, misconstruals, and errors of fact or judgment are the sole responsibility of the author.

[1] Related work making use of Harmonic Grammar includes (Keller 2000, Keller 2006, Pater et al. 2007b). Related work making use of non-linear numeric optimization includes (Johnson 2002).

[2] Pater et al. (2007a) use the term cumulative interaction; Keller and Asudeh (2002) use the term **cumulative effect** to mean essentially the same thing.

(1) Major points of this paper.

a. Many cases permitting OT and HG to diverge can be understood in terms of disproportional constraint violation differences: the violation difference ratios for conflicting constraints differs across candidate comparisons.

b. Failure to exhibit cumulative interaction is not limited to instances of “Anything Goes” competitions; there are competitions which are not “Anything Goes”, but do exhibit proportional violation differences, and do not permit cumulative interactions.

c. Efforts to avoid cumulative interactions in some comparisons can introduce cumulative interactions into other comparisons.

d. Imposing a fixed bound on the number of violations a constraint can assess does not eliminate typologically problematic patterns in HG, it only limits the number of variations.

Many parts of this paper are summary and synthesis of insights originally presented elsewhere, especially in (Legendre et al. 2006, Pater et al. 2007a, Prince and Smolensky 1993/2004, Prince 2002), with a few new observations along the way. I make particularly extensive use of the investigation in PBP; many of the examples are either taken from or adapted from PBP. The construct of violation difference ratios makes it possible to tie together many of these prior insights in a particularly understandable way.

2 Basics of Lexicographic and Linear Numeric Optimization

2.1 OT and HG

Both OT and HG optimize over a space of candidates, such that the optimal candidate is at least as harmonic as each competitor. In both theories, constraint violations (on their own) always have a negative effect on harmony: given two candidates which differ only in the degree of violation of a single constraint, the candidate with fewer violations always wins. Given this property, more must be said only when constraints **conflict** on a comparison between two candidates: when one constraint assigns fewer violations to one candidate, while the other constraint assigns fewer violations to the other candidate. OT and HG differ in their methods of resolving conflicts between constraints.

OT uses lexicographic ordering. A total order (ranking) is imposed on the constraints, and the highest constraint in the order with a preference between two candidates decides the comparison, in favor of the candidate with fewer violations. The order of the constraints resolves conflicts between them, and determines the relative harmony of any pair of candidates.

HG uses linear numeric ordering: it assigns harmony values to candidates that are linearly weighted sums of the violations of the constraints. Each constraint is assigned a numeric weight, and the harmony of a candidate is the sum, over all of the constraints of the system, of the weight of that constraint multiplied by the number of violations of that constraint. The resulting harmony values are also numbers, and the standard order on numbers determines the relative harmony of any pair of candidates.

For HG, this paper will observe the following conventions. Constraint weights are strictly positive numbers. Constraint violation counts are non-negative integers. The harmony of a candidate is the sum, over the constraints, of the negative of the product of the weight of a constraint by the number of violations of that constraint. Harmony values for candidates are thus always non-positive numbers, and the maximum harmony will be the highest harmony value. Note that the highest harmony value will have the smallest magnitude. The weight of constraint C will be denoted w(C), and number of violations of C assessed to candidate **cand** will be denoted C(**cand**). Given these conventions, the harmony assigned to a candidate by an HG grammar is shown in (2).

(2) $$H(cand) = \sum_{C \in Con} -C(cand) * w(C)$$

For both OT and HG, the state of affairs in which **a** is more harmonic than **b** will be denoted **a**≻**b**. A comparison between two candidates without an a priori commitment as to which is more harmonic will be denoted **a**~**b**.

2.2 When the Many Outweigh the Few

A simple difference between OT and HG can be found in a situation where one violation of one constraint conflicts with a large number of violations of another constraint. Consider the tableau in Table 1 (the tableau conventions for HG arguments largely follows PBP). Two candidates, **a** and **b**, compete. Candidate **b** has one violation of C1, and candidate **a** has *x* violations of C2; assume here that *x*>1. In OT, **a**≻**b** under the ranking C1 ≫ C2. In HG, C1 has weight w(C1) and C2 has weight w(C2), and **a** only has a chance of being more harmonic than **b** if w(C1) > w(C2).

Table 1 One violation of C1 conflicts with *x* violations of C2

Weight	w(C1)	w(C2)	*H*
	C1	C2	
a	0	*x*	-(*x**w(C2))
b	1	0	-(1*w(C1))

In OT, If C1 ≫ C2, then **a**≻**b**, no matter how large *x* is. No value of *x* will yield **b**≻**a** with this ranking.

In HG, the harmony of **a** is H(**a**) = -(*x**w(C2)). H(**b**) = -(1*w(C1)). **b**≻**a** will hold whenever -(1*w(C1)) > -(*x**w(C2)), or equivalently w(C1) < *x**w(C2) . This will happen when *x*>w(C1)/w(C2). A sufficient number of violations of the smaller-weighted C2 can outweigh a violation of the larger-weighted C1.

Thus, no matter what values for w(C1) and w(C2) are adopted, there will be violation patterns on which the OT and HG grammars diverge. When w(C1) > w(C2), no numeric weights can achieve the OT pattern of one violation of C1 being less harmonic than an arbitrary number of violations of C2, and the analogous result applies when w(C2) > w(C1). Linear numeric ordering cannot implement lexicographic ordering exactly, an observation dating back to the original, defining work on OT (Prince and Smolensky 1993/2004, p. 236).[3]

The same interactions can also be used to show that lexicographic ordering cannot implement linear numeric ordering exactly. This is illustrated in Table 2 by choosing particular weights: w(C1) = 1.5, w(C2) = 1.

[3] PBP:7 appears to assert exactly the opposite: “any linguistic pattern that can be analyzed with some set of ranked constraints can also be analyzed in terms of weightings of the same constraints (Prince and Smolensky 2004:236; Prince 2002a).” However, Pater (2007, p.c.) suggests an alternative interpretation of PBP:7, in which ‘linguistic pattern’ refers specifically to finite sets of specific forms. If attention is restricted to a finite set of forms, then there is a largest number of violations among the forms for each constraint, and weights can be found which mimic lexicographic ordering up to that degree of violation.

Table 2 HG can have a and d be optimal simultaneously; OT cannot.

Weight	1.5	1	*H*
	C1	C2	
a	0	1	-1
b	1	0	-1.5
c	0	2	-2
d	1	0	-1.5

In Table 2, **a**≻**b**, but **d**≻**c**. A threshold on the tradeoff between C1 and C2 has been set by HG: one violation of C2 is better than one violation of C1, but two violations of C2 are worse. No ranking of C1 and C2 in OT can achieve this pattern. The example in Table 1 shows that for any selection of weights, such a threshold will exist, where a sufficiently large number of violations of the smaller-weighted constraint will outweigh one violation of the larger-weighted constraint.

This is a cumulative interaction: the pattern is due in part to the fact that constraints accumulate violations. OT is sensitive to the fact of two candidates having different numbers of violations, but not to the degree of difference. HG is sensitive to the degree of difference in the numbers of violations between two candidates.

A key observation is that the cumulative interactions are due not to the absolute numbers of violations, but the difference in the numbers of violations. This can be illustrated by increasing all of the violation counts in Table 2 by 100, yielding the tableaux in Table 3. The effect of this is to reduce the harmony of all candidates by 250 (-150 for C1, -100 for C2). Candidate **a** still has 1 less violation of C1 than **b**, and 1 more violation of C2. This reflects a property shared by OT and HG: it is the differences in numbers of violations on a given constraint that matter.

Table 3 Differences in the numbers of violations, not absolute numbers, are what matter.

Weight	1.5	1	*H*
	C1	C2	
a	100	101	-251
b	101	100	-251.5
c	100	102	-252
d	101	100	-251.5

The legitimacy, within OT, of attending to only the differences in the number of violations is the ‘Cancellation Lemma’ (Prince and Smolensky 1993/2004, p. 164). The significance only of the **violation differences** between competitors is just as true in HG as it is in OT. It is the (in)sensitivity to the magnitude of the differences that distinguishes them. The consequence of the distinction is that OT exhibits certain patterns that cannot be realized in HG, and HG exhibits certain patterns that cannot be realized in OT.

2.3 Simple and Collective Harmonic Bounding

Further insight into the similarities and differences between OT and HG can be gained by considering the two varieties of harmonic bounding that occur in OT.

A simple harmonic bounding relationship holds between two candidates if one candidate, the bounder, has no more violations than the other candidate, the boundee, for any of the constraints, and the bounder has strictly fewer violations on at least one of the constraints. A simple illustration is given in Table 4, where candidate **b** is harmonically bound by candidate **a**.

Table 4 Candidate b is simply harmonically bound by candidate a.

	C1	C2	C3
a	0	1	1
b	1	2	1

As noted by Prince (2002) (and also in PBP), candidates which are simply harmonically bound cannot be optimal in either OT or HG. In OT, no matter what ranking is imposed on the constraints, the highest one with a preference will prefer the bounder over the boundee. In HG, no matter what weights are assigned to the constraints, every constraint that distinguishes the two candidates will lower the harmony of the boundee more than that of the bounder. In Table 4, candidate **b** will be more harmonic than candidate **a** no matter what. Simple harmonic bounding relations hold no matter what ranking or weighting is assigned to the constraints.

Simply harmonically bound candidates cannot succeed because they pose no constraint conflict. In both OT and HG, ranking and constraint weighting serve to adjudicate conflicts between constraints. When there are no conflicts in a comparison, there is no work to be done by conflict resolution. The constraints themselves still do work; differentiating the importance between them does not. It is worth noting that, for this reason, simple harmonic bounding comparisons provide no information about the grammar (to analyst or learner). They are compatible with any ranking and any weighting, and thus provide no information.

The situation is rather different with collective harmonic bounding (Samek-Lodovici and Prince 1999). Collective harmonic bounding in OT is illustrated in Table 5. When C1$\gg$C2, **a**$\succ$**b**, and thus **b** cannot be optimal. When C2$\gg$C1, **c**$\succ$**b**, and thus **b** cannot be optimal. Candidates **a** and **c** conspire to prevent **b** from being optimal under any ranking.[4]

Table 5 Candidate b is collectively harmonically bound in OT by candidates a and c.

	C1	C2
a	0	3
b	1	1
c	2	0

Prince (2002) has noted that collective harmonic bounding relationships in OT are not always preserved in HG. In HG, **b** is optimal when: w(C1)+w(C2) < 3*w(C2) AND w(C1)+w(C2) < 2*w(C1). Simplifying to the violation differences yields the two conditions given in (3).

[4] An example which is similar in structure is discussed in (Pater et al. 2007b).

(3) **b** will be more harmonic than **a** and **c** when these two conditions are satisfied.

a. w(C1) < 2*w(C2) (**b**≻**a**)

b. w(C2) < w(C1) (**b**≻**c**)

These conditions are satisfied when, for example, w(C1) = 1.4 and w(C2) = 1. The resulting harmony values of the candidates are shown in Table 6.

Table 6 Candidate b, collectively harmonically bounded in OT, can be optimal in HG.

Weight	1.4	1	*H*
	C1	C2	
a	0	3	-3
b	1	1	-2.4
c	2	0	-2.8

The collective harmonic bounding situation here is really just another example of the kind of situation described in the previous section that is realizable in HG but not OT. There are two comparisons, each sporting conflict between two constraints. Each constraint prefers the winner in one comparison and the loser in the other. The collective harmonic bounding is a special case where the two comparisons have the same winner, and thus are both part of a single competition.

Thus, the same formal distinction between OT and HG can give rise to combinations of optimal forms that cannot be simultaneously optimal in OT but can in HG (Table 2), and individual forms that cannot be optimal under any grammar in OT but can in HG (Table 5).

2.4 Gang Effects

Cumulative interactions are not restricted to pairwise constraint interactions. Violations across multiple constraints can combine to overcome those of another constraint, as illustrated in Table 7. A single violation of either C2 or C3 will be tolerated to avoid a single violation of C1. But the competition between **e** and **f** poses a conflict between C1 and both C2 and C3, and the sum of the weights of C2 and C3 is greater than the weight of C1. This is a **gang effect**: constraints C2 and C3 may be viewed as a gang that can collectively outweigh C1.

Table 7 f beats e, because C2 and C3 combine to overcome C1.

Weight	1.5	1.1	1	*H*
	C1	C2	C3	
☞ **a**			1	-1
b	1			-1.5
☞ **c**		1		-1.1
d	1			-1.5
e		1	1	-2.1
☞ **f**	1			-1.5

In some respects, the effect here is similar to that illustrated in Table 2. The violations of C2 and C3 are like two violations of a single constraint for **f**≻**e**, because for that comparison both constraints **conflict** with C1. However, because C2 and C3 are separate constraints, lexicographic ordering has more freedom in distinguishing **e** and **f**: **f**≻**e** can be accounted for with either C2≫C1 or C3≫C1.[5] The distinction between lexicographic and linear numeric ordering arises when both ranking options (C2≫C1 and C3≫C1) are blocked by other comparisons, as is the case in Table 7. Linear numeric ordering also has more freedom when C2 and C3 are separate constraints, because different weights can be assigned to the constraints, but that additional freedom is not relevant to this particular case.

One interesting consequence of this is another qualitative difference between OT and HG. In OT, comparisons between candidates with non-identical violation profiles are always resolved. In HG, candidates with non-identical violation profiles can tie (have identical harmony values), as illustrated in Table 8. Note that this phenomenon is **not** necessarily a consequence of any kind of constraint tying: in Table 8, the constraints all have distinct weights.

Table 8 Non-identical violation profiles can tie in harmony.

Weight	4	3	2	1	*H*
	C1	C2	C3	C4	
a	0	1	1	0	-5
b	1	0	0	1	-5

3 Violation Differences

PBP use interactions between NOCODA and MAX to demonstrate two cases: a case where OT and HG cannot diverge, and a case where they do. In this section, both cases are reviewed. The two cases arise from two different kinds of linguistic accumulation. It will be shown that the two cases can be understood and related to each other via the concepts of proportional and disproportional violation differences. Here, NOCODA assigns one violation to each syllable with a coda, and MAX assigns one violation to each input segment that lacks an output correspondent.

3.1 *Independent interactions: multiple codas*

NOCODA and MAX interact in cases where candidates offer a choice between putting a consonant in a coda position and deleting the consonant, as shown in Table 9 (from PBP:8).

Table 9 NOCODA and MAX conflict.

/bat/	NOCODA	MAX
ba	0	1
bat	1	0

When forms have more than one plausible coda position, multiple instances of the conflict arise; this is illustrated in Table 10 (from PBP:8), using a form with two plausible coda sites.

[5] Reflecting the general OR/AND logic of OT comparison: at least one of the constraints preferring the winner must dominate all of the constraints preferring the loser.

Table 10 Anything Goes: the same resolution must be taken for both conflicts, so [banta] cannot be optimal.

/bantat/	NoCoda	Max
bata	0	2
banta	1	1
bantat	2	0

Each potential coda involves a separate trade-off between NoCoda and Max. Each interaction is independent. As demonstrated in PBP, if we examine the inequalities that must be satisfied for candidate [banta] to be more harmonic than the other two candidates, we see that they are not jointly satisfiable (PBP:8-9 uses slightly different competitions, but the point is exactly the same). Such inequalities are termed **weighting conditions** in PBP. The weighting conditions for this example are shown in (4).

(4) Unsatisfiable: the weight of NoCoda cannot be simultaneously less than and greater than the weight of Max.

a. 1*w(NoCoda) < 1*w(Max) (banta≻bata)

b. 1*w(Max) < 1*w(NoCoda) (banta≻bantat)

This is a form of an "Anything Goes" competition (Prince 2002). A candidate set with given constraints is said to be Anything Goes with respect to a given ranking if the optimum selected by OT using that ranking will also be selected by HG using any assignment of positive weights such that the order of the sizes of the weights matches the ranking. In fact, this candidate set with these constraints is "Anything Goes Factorial": **any** assignment of weights to the constraints with w(NoCoda)>w(Max) will have the same effect as NoCoda≫Max in OT, and any assignment of weights with w(Max)>w(NoCoda) will have the same effect as Max≫NoCoda in OT. With such sets of violation profiles, it is only the linear order of the weights that matters (exactly as in OT); the magnitudes of the differences between the weights is irrelevant.[6]

"Anything Goes" situations are a special case of the circumstances in which cumulative interactions are not possible. A more general class of cases involves what I will call **proportional violation differences**: the differences in the number of violations for conflicting constraints are proportional across comparisons. In Table 10, NoCoda and Max conflict on comparison [banta]~[bata], and both constraints have a violation difference of 1. Both constraints also conflict on comparison [bantat]~[bata], but in this comparison both have a violation difference of 2. The violation differences are proportional across the comparisons: when the difference for one constraint doubles, so does the other.

One way to think of violation differences is in terms of trade-offs between violations of conflicting constraints.[7] In the example in Table 10, the trade-off is one-to-one for the comparisons [banta]~[bata] and [banta]~[bantat]: each elimination of one violation of NoCoda incurs exactly one additional violation of Max, and vice-versa. For the comparison [bata]~[bantat], the trade-off is 2-to-2, which is equivalent to 1-to-1.

Linguistically, the constraint violation differences are proportional in the cases examined so far because each potential site poses an independent instance of the same choice between a single consonant coda and no coda at all. The trade-offs are independent; that is the point of the example in PBP. It is the

[6] A complete analysis of the conditions under which comparisons, competitions, and typologies qualify as "Anything Goes" can be found in the paper of the same name (Prince 2002).
[7] Such interactions are described in terms of "traded off" violations in PBP:12.

independence of these repeated configurations that keeps the violation differences proportional: each potential coda site represents a separate conflict between one violation of NOCODA and one violation of MAX.

Proportional violation differences can be more clearly expressed with the help of a useful bit of notation: let C(**a**~**b**) be the magnitude of the difference in the number of violations of constraint C between candidates **a** and **b**.

(5) Restating (4) in terms of violation differences.

a. NOCODA(banta~bata)*w(NOCODA) < MAX(banta~bata)*w(MAX)

b. MAX(banta~bantat)*w(MAX) < NOCODA(banta~bantat)*w(NOCODA)

Manipulating the expressions in (5) yields equivalent expressions that explicitly contain ratios of the violation differences of the comparisons.

(6) The inequalities relate ratios of violation differences to ratios of constraint weights.

a. $$\frac{\text{w(NOCODA)}}{\text{w(MAX)}} < \frac{\text{MAX(banta\~{}bata)}}{\text{NOCODA(banta\~{}bata)}}$$

b. $$\frac{\text{MAX(banta\~{}bantat)}}{\text{NOCODA(banta\~{}bantat)}} < \frac{\text{w(NOCODA)}}{\text{w(MAX)}}$$

Satisfying the inequalities requires that the ratio of the constraint weights, w(NOCODA) / w(MAX), be less than the ratio (trade-off) of violation differences for one comparison (banta~bata), and greater than the ratio of violation differences for the other comparison (banta~bantat). These cannot be simultaneously satisfied when the ratios of violation differences for the two comparisons are equal (you cannot fit anything in between 1 and 1). If the violation differences are proportional across two comparisons, the ratios of violation differences for those comparisons will be equal. In Table 10, this is the case.

(7) $$\frac{\text{MAX(banta\~{}bata)}}{\text{NOCODA(banta\~{}bata)}} = \frac{\text{MAX(banta\~{}bantat)}}{\text{NOCODA(banta\~{}bantat)}} = \frac{1}{1} = 1$$

In general, preventing cumulative interactions does not require that the violations of conflicting constraints trade off 1 for 1 everywhere; it requires that they trade off proportionally, for all comparisons in the situation. The ratio of violation differences needn't be 1, it just needs to be constant. In Table 11, candidate **b** cannot be optimal under any weighting of the constraints: as shown in (9), the inequalities expressing the conditions for **b** to be more harmonic than **a** and **c** cannot be simultaneously satisfied. The ratio of violation differences C2/C1 for all comparisons is equal to 2/1, as shown in (8) (the violation trade-offs C2-to-C1 are all equivalent to 2-to-1).

Table 11 Two violations of C2 trade off against one violation of C1.

Weight	w(C1)	w(C2)
	C1	C2
a	0	4
b	1	2
c	2	0

(8) The violation differences are proportional across the comparisons.

a. $\frac{C2(a\sim b)}{C1(a\sim b)} = \frac{2}{1}$

b. $\frac{C2(b\sim c)}{C1(b\sim c)} = \frac{2}{1}$

(9) Unsatisfiable: the ratio of the constraint weights cannot be simultaneously less than and greater than the ratio of the violation differences.

a. $\frac{w(C1)}{w(C2)} < \frac{2}{1}$ (**b**≻**a**)

b. $\frac{2}{1} < \frac{w(C1)}{w(C2)}$ (**b**≻**c**)

The example in Table 11 is **not** an “Anything Goes” competition with respect to the ranking C1≫C2. This is illustrated in Table 12. The assigned weights select **c** as the optimal candidate in HG, while ranking the constraints in the same order as the order of the weights (C1≫C2) selects **a** as optimal in OT. However, this reversal simply exchanges one OT-accessible solution for another one: the weights indicated in Table 12 are equivalent to the OT ranking C2≫C1 for this particular set of candidates. Because the violation differences are proportional, any numeric weights which distinguish all of the candidates will select a winner consistent with some OT ranking.

Table 12 Not “Anything Goes”: OT picks candidate a, HG picks candidate c.

Weight	1.0	0.6	*H*
	C1	C2	
a	0	4	-2.4
b	1	2	-2.2
c	2	0	-2.0

A qualification should be made here for the special case in which the ratio of the weights is exactly equal to the ratio of the violation differences. This will be true when w(C1) = 1.0 and w(C2) = 0.5. In that instance, all three candidates will tie for optimality. This is an instance of candidates with non-identical violation profiles tying in harmony (section 2.4). Note that this circumstance cannot be avoided simply by stipulating that all constraints have non-identical weights; 1.0 is not identical to 0.5.

The violation difference ratios are essentially a restructuring of the weighting conditions described in PBP. Expressing things in terms of violation difference ratios makes the equivalence classes of constraint interactions particularly transparent: 1/1 is equivalent to 2/2, and so forth. It makes it easy to see how the same relationships can hold for trade-offs that aren’t equivalent to 1/1: 2/1 is equivalent to 4/2, 3/2 is equivalent to 6/4, etc. It also makes it easy to understand cases that are not “Anything Goes”, but do not stray outside the typological bounds of OT: they exhibit proportional violation differences.

3.2 Disproportional violation differences: clusters in a coda

The interaction between NoCoda and Max is rather different when we consider the possibility of a coda with more than one consonant. Table 13, based upon tableaux from PBP:20, shows two competitions, the first with one potential coda consonant and the second with two potential coda consonants. The second competition involves two potential violations of Max but only one potential violation of NoCoda. These violation profiles are identical to those in Table 2: an OT ranking of the constraints cannot make candidates **a** and **d** simultaneously optimal, but an HG weighting can.

Table 13 HG can impose a "minimum coda cluster size".

	Weight	1.5	1	*H*
	/bat/	NoCoda	Max	
☞ **a**	ba	0	1	-1
b	bat	1	0	-1.5
	/bant/			
c	ba	0	2	-2
☞ **d**	bant	1	0	-1.5

As is shown in PBP with this example, the asymmetry in the violation trade-off in the second competition, away from the 1-to-1 trade-off of the first competition, makes the cumulative interaction possible. This situation can be expressed in terms of the disproportional violation differences between the two comparisons. The ratios of violation differences are shown in (10): for the first competition it is one violation of Max over one violation of NoCoda, while for the second competition it is two violations of Max over one violation of NoCoda. The unequal ratios are the result of NoCoda only being violated once by a coda no matter how large, while Max incurs a separate violation for every potential coda consonant that is deleted.

(10) The violation differences in Table 13 are disproportional.

a. $\frac{\text{MAX}(\text{a\~b})}{\text{NOCODA}(\text{a\~b})} = \frac{1}{1} = 1$ (**a**≻**b**)

b. $\frac{\text{MAX}(\text{c\~d})}{\text{NOCODA}(\text{c\~d})} = \frac{2}{1} = 2$ (**d**≻**c**)

The disproportional violation differences provide the opening for an HG analysis to make **a** and **d** simultaneously optimal, as illustrated in (11). Because the two violation difference ratios aren't equal, one can define a ratio of weights of w(NoCoda) to w(Max) that falls in-between the two violation difference ratios. One such solution consists of the weights shown in Table 13: the weight ratio is 1.5 / 1 = 1.5 (as a ratio of integers, 3 to 2), which falls in-between 1 and 2.

(11) Satisfiable: the weight ratio can fall between 1 and 2.

a. MAX(**a**~**b**)*w(MAX) < NOCODA(**a**~**b**)*w(NOCODA) (**a**≻**b**)

b. $1 = \frac{\text{MAX(a~b)}}{\text{NOCODA(a~b)}} < \frac{\text{w(NOCODA)}}{\text{w(MAX)}}$ (**a**≻**b**)

c. NOCODA(**a**~**b**)*w(NOCODA) < MAX(**a**~**b**)*w(MAX) (**d**≻**c**)

d. $\frac{\text{w(NOCODA)}}{\text{w(MAX)}} < \frac{\text{MAX(a~b)}}{\text{NOCODA(a~b)}} = 2$ (**d**≻**c**)

Linguistically, the effect here is that of a "minimum coda cluster". Potential coda clusters that are below the minimum size will be deleted, while those at or above the minimum size will be retained. For the weights shown in Table 13, the minimum coda is two segments. A sufficient increase in the weight of NOCODA will increase the size of the minimum coda cluster. The minimum coda cluster size will be the smallest multiple of the weight of MAX that is larger than the weight of NOCODA.

As discussed above for Table 2, the patterns for OT and HG are divergent. For OT, if NOCODA dominates MAX then there will be no codas, no matter how potentially large. For HG, there will always be a minimum coda cluster size, no matter how large the weight of NOCODA is relative to the weight of MAX.

3.3 Combining the patterns

Of course, both kinds of interactions can occur within single competitions. This is shown in Table 14.

Table 14 The interactions combine: a failure to realize OT-style collective harmonic bounding.

	Weight	1.5	1	*H*
	/bat/	NOCODA	MAX	
☞ **a**	ba	0	1	-1
b	bat	1	0	-1.5
	/bant/			
c	ba	0	2	-2
☞ **d**	bant	1	0	-1.5
	/bantant/			
e	bata	0	3	-3
☞ **f**	batant	1	1	-2.5
g	bantant	2	0	-3

The bottom tableau in Table 14 is an instance of collective harmonic bounding for OT; the violation profiles are identical to those of Table 5. The cumulative interaction in the collective harmonic bounding is a consequence of the disproportional violation differences. The comparison between **f** and **e** is formally equivalent to the comparison between **d** and **c**: the trade-off is two violations of MAX to one violation of NOCODA. The comparison between **f** and **g** is formally equivalent to the comparison between **a** and **b**: the trade-off is one violation of MAX to one violation of NOCODA. The minimum coda cluster is enforced uniformly across longer forms.

Note the minor but highly significant difference between the tableau for /bantat/ in Table 10 and the bottom tableau, for /bantant/, of Table 14. The two differ by exactly one violation: the first candidate in Table 10 has one less violation of MAX. The margin between Anything Goes and cumulative interaction can be slim indeed. Non-OT cumulative interactions can arise when the differences in the number of violations for conflicting constraints are not proportional across candidate comparisons. Put another way, cumulative interactions can arise when the ratios between violation differences of conflicting constraints are not equal across comparisons.

To avoid the cumulative interaction here, one needs to avoid the disproportional violation differences. For this particular case of coda clusters, a possibility suggested in PBP:21 is to change the definition of NOCODA so that it assesses a separate violation for each coda consonant. I will call this constraint NOCODA-SEG. The tableaux resulting from the replacement of NOCODA with NOCODA-SEG are shown in Table 15.

Table 15 Cumulative interaction blocked: all comparisons have proportional violation differences

	/bat/	NOCODA-SEG	MAX
a	ba	0	1
b	bat	1	0
	/bant/		
c	ba	0	2
d	bant	2	0
	/bantant/		
e	bata	0	3
f	batant	2	1
g	bantant	3	0

Now, candidate **f** is not a possible optimum, and candidates **a** and **d** cannot be simultaneously optimal. The constraint violation difference ratio in the comparison between **f** and **e** is 2/2, which is equal to 1. The ratio in the comparison between **f** and **g** is 1/1, also equal to 1. The violation differences between the conflicting constraints are proportional across the comparisons.

3.4 The Limits of Disproportionality

The disproportional violation differences displayed in Table 13 make it possible to have combinations of comparisons that are satisfiable in HG but not in OT. However, there are limits on the combinations that HG can satisfy. In Table 13, there are two comparisons: **a**~**b** and **c**~**d**. Each grammar must select exactly one candidate from each comparison as the more harmonic for that comparison. Both OT and HG can select the set {**a**,**c**} as the more harmonic candidates; both can satisfy **a**≻**b** and **c**≻**d**. Both OT and HG can select the set {**b**,**d**} as the more harmonic candidates; both can satisfy **b**≻**a** and **d**≻**c**. As demonstrated above, the disproportional violation differences allow HG, but not OT, to select {**a**,**d**} as the more harmonic candidates; HG can satisfy **a**≻**b** and **d**≻**c**.

Neither OT nor HG can select {**b**,**c**} as the more harmonic candidates; neither can simultaneously satisfy **b**≻**a** and **c**≻**d**. This is exactly the opposite of (**a**≻**b** and **d**≻**c**), the conditions for which were described in (11). The conditions for (**b**≻**a** and **c**≻**d**) can be obtained by simply reversing the conditions for (**a**≻**b** and

d≻**c**); this is shown in (12). In the terms of PBP, the weighting conditions cannot be simultaneously satisfied, analogous to the case in (4). No number exists which is simultaneously less than 1 and greater than 2.

(12) Not satisfiable in HG or OT: the weight ratio cannot be less than 1 and greater than 2.

a. $$1 = \frac{\text{MAX(a\textasciitilde b)}}{\text{NOCODA(a\textasciitilde b)}} > \frac{\text{w(NOCODA)}}{\text{w(MAX)}}$$ (**b**≻**a**)

b. $$\frac{\text{w(NOCODA)}}{\text{w(MAX)}} > \frac{\text{MAX(a\textasciitilde b)}}{\text{NOCODA(a\textasciitilde b)}} = 2$$ (**c**≻**d**)

The comparison (**b**≻**a**) forces the weight ratio to be less than 1, which requires that w(NoCoda) < w(Max); Max must have a larger weight than NoCoda. Given that constraint order, the comparison (**c**≻**d**) then makes the impossible demand of one violation of a lower-weighted constraint overcoming two violations of a higher-ranked constraint. In fact, the comparison (**c**≻**d**) is Anything Goes with respect to the constraint ranking Max ≫ NoCoda: any assignment of weights such that w(Max)>w(NoCoda) in HG will yield the same judgment as OT with the given ranking.[8] The conditions under which a comparison is Anything Goes with respect to a specific ranking are more complex than a simple comparison of the total numbers of violations of the two candidates when more than two constraints are involved: see (Prince 2002) for the details.

Disproportional violation differences between two comparisons creates potential for patterns which can be realized in HG but not in OT. It does not render the two comparisons independent in HG.

4 Accumulation Across Constraints (Gang Effects)

Gang effects, such as those described in section 2.4, are a step towards the general case in which conflicts can be between more than two constraints, and constraints which conflict on one comparison do not conflict on another. The simple analysis in terms of (dis)proportional violation differences works when the same (sets of) constraints conflict on all relevant comparisons. This is due to the nature of lexicographic ordering: if, across several comparisons, exactly the same constraints prefer the winners and exactly the same constraints prefer the losers, lexicographic ordering cannot tell the difference between the comparisons, while linear numeric ordering can. If one or more constraints changes its conflict relationships with other constraints across comparisons, as is the case in Table 7, then further investigation is required to determine which combinations of preferences are and are not realizable by lexicographic ordering.

While the fully general case is complex, greater understanding of gang effects can still be gained by treating the constraints of a "gang" as if they were a single constraint for purposes of evaluating (dis)proportional violation differences. When conflicting gangs of constraints exhibit disproportional violation differences, they create opportunities for cumulative interactions.

I first give an example in which gang effects **fail** to occur. Adapting an example from PBP:11, consider an attempt to derive coda-specific restrictions on voiced obstruents from separate markedness constraints against codas and voiced obstruents. If only the constraints Ident-Voice, NoCoda, and *VoiceObs are considered, this attempt fails. The key competitions are shown in Table 16.

[8] The candidate set {**c**,**d**} is not Anything Goes Factorial: if we choose NoCoda≫Max, a cumulative effect is possible, as shown in section 3.2, with OT ordering **c**≻**d** but HG ordering **d**≻**c**.

Table 16 Gang effects denied: NoCoda does not distinguish [pa] from [ba], nor [pat] from [pad].

	IDENT-VOICE	NOCODA	*VOICEOBS
/ba/			
pa	1		
ba			1
/pat/			
pat		1	
pad	1	1	1
/pad/			
pat	1	1	
pad		1	1

There is no difficulty in finding weights for the constraints such that sum of w(NOCODA) and w(*VOICEOBS) is greater than w(IDENT-VOICE). The difficulty is that such weights won't ensure that coda obstruents are voiceless. Further, any weights ensuring that coda obstruents are voiceless will also ensure that onset obstruents are voiceless (/ba/→[pa]); no coda-specific pattern is obtained. This is a consequence of the fact that voiced and voiceless codas are both still codas, and thus constitute violations of NOCODA. NOCODA does not distinguish the shown candidates for any input: the violation difference between the candidates on NOCODA is zero, for all comparisons. NOCODA is effectively absent.

If we group NOCODA and *VOICEOBS into a gang, we find proportional violation differences between (NOCODA, *VOICEOBS) and IDENT-VOICE: the trade-off is 1-to-1 for both comparisons in Table 16. In this set of comparisons, the gang doesn't get to behave like one: NOCODA is not conflicting with anything, and thus isn't worthy of gang membership. The "gang" is an illusion.

A cumulative interaction could be achieved if a constraint could be found that had disproportional violation differences with the gang (NOCODA, *VOICEOBS). Such a constraint is close at hand if we are willing to slightly expand the considered sets of candidates. Consider the addition of the constraint MAX, and the inclusion of candidates that delete input consonants. This is shown in Table 17. This example directly parallels one discussed by Pater (2007, p. 6) involving the interaction of IDENT-BACK, MAX, and constraints involved in vowel harmony. The indicated weights achieve the pattern described in (13).

(13) A coda-specific voicing pattern, achieved via gang effects

- voiced (and voiceless) obstruents are retained in onset position (neither devoiced nor deleted).
- voiceless obstruents are retained in coda position (not deleted).
- voiced obstruents are deleted in coda position.

In the competition for input /ba/, *VOICEOBS alone cannot overcome MAX. In the competition for input /pat/, *NOCODA alone cannot overcome MAX. In the competition for input /pad/, the gang of *VOICEOBS and NOCODA combined overcomes MAX, causing [pad] to lose to [pa]. There is no lexicographic ordering of these same constraints in OT that can achieve this pattern.

Table 17 Gang effects achieved: coda-specific effects enforced via deletion.

	Weight	2	1.5	1.4	1	*H*
		MAX	IDENT-VOICE	NOCODA	*VOICEOBS	
	/ba/					
	pa		1			-1.5
☞	ba				1	-1
	a	1				-2
	/pat/					
☞	pat			1		-1.4
	pad		1	1	1	-3.9
	pa	1				-2
	/pad/					
	pat		1	1		-2.9
	pad			1	1	-2.4
☞	pa	1				-2

This gang effect is possible because deleting a voiced obstruent in coda position removes violations of both members of the gang (NOCODA, *VOICEOBS). (NOCODA, *VOICEOBS) and MAX have disproportional violation differences. The trade-off is 2-to-1 in the comparison between [pad] and [pa] for input /pad/. The trade-off is 1-to-1 for the comparisons for inputs /ba/ and /pat/. The disproportional violation differences open the door to cumulative interactions. In the case of gang effects, the accumulation is across constraints, acting in concert.

The cumulative interaction between (NOCODA, *VOICEOBS) and MAX has violation profiles that are identical to those in Table 7. The relevant profiles are shown in Table 18.

Table 18 The key interactions between MAX and (NOCODA,*VOICEOBS)

		MAX	NOCODA	*VOICEOBS
	/ba/			
☞	ba			1
	a	1		
	/pat/			
☞	pat		1	
	pa	1		
	/pad/			
	pad		1	1
☞	pa	1		

This is another example of the potential of HG to make interesting predictions via cumulative interactions. Given this set of constraints and candidates, the prediction is that a ban on voiced obstruents in codas can be enforced via segment deletion but not via devoicing. The substance of this prediction is a bit awkward; the empirically observed pattern about enforcement of voiced coda bans is exactly the reverse (Lombardi 2001).[9] But restricting voicing analyses to just these constraints is implausible anyway. This fact shouldn't obscure the more general point that cumulative interactions in HG cannot "do everything". However, it should be noted that Local Harmonic Serialism, discussed in PBP, is related to serialist analyses proposed by McCarthy (2006a) that make much more empirically respectable predictions about "repair" asymmetries. See (Pater 2007) for discussion.

5 Multiple Shifting Interactions

5.1 *Geminate devoicing in Japanese loanwords*

PBP discusses a cumulative interaction, also involving gang effects, in an analysis of geminate devoicing in Japanese loanwords (PBP:5-7). Simplifying somewhat, the pattern of interest (Kawahara 2006, Nishimura 2003) is shown in (14); the data come from (Kawahara 2006). In this pattern, non-geminate voiced obstruents are permitted, but a voiced geminate obstruent is permitted only in the absence of another voiced obstruent.

(14) Geminate devoicing in Japanese loanwords

/bobu/ → bobu 'Bob'

/webbu/ → webbu 'web'

/doggu/ → dokku 'dog'

/deibiddo/ → deibitto 'David'

[9] Pater (2007) notes that the same kind of interaction in his vowel harmony example also yields an unattested pattern.

PBP describe this pattern as geminate devoicing in a Lyman's Law environment (see also (Nishimura 2003)), in reference to the restriction in Japanese that a word contain only a single voiced obstruent (a restriction that is not enforced in the loanwords). What is of interest here is PBP's HG analysis of this pattern, an analysis that exploits cumulative interaction. The analysis is illustrated in Table 19. In this analysis, *VCE-GEM assesses a separate violation to each voiced geminate, and *2-VOICE allows only one voiced obstruent in a word.

Table 19 Geminates are devoiced only in the presence of another voiced obstruent

	Weight	1.5	1	1	*H*
		IDENT-VOICE	*VCE-GEM	*2-VOICE	
	/bobu/				
☞	bobu			1	-1
	bopu	1			-1.5
	/webbu/				
☞	webbu		1		-1
	weppu	1			-1.5
	/doggu/				
	doggu		1	1	-2
☞	dokku	1			-1.5

The comparisons for forms [bobu] and [webbu] ensure that the weights of *VCE-GEM and *2-VOICE must both be less than the weight of IDENT-VOICE. In the comparison for [dokku], the combined weights of *VCE-GEM and *2-VOICE are greater than the weight of IDENT-VOICE, resulting in the devoiced geminate. This is another example of a gang effect: the gang (*VCE-GEM, *2-VOICE) conflicts with IDENT-VOICE. The violation profiles in Table 19 are identical to those in Table 7.

The analysis exploits cumulative interactions. In doing so, it accounts for [dokku] using only independently motivated constraints (*2-VOICE is motivated by Lyman's law, and *VCE-GEM is motivated by the ban on voiced geminates in Japanese), without the need for an additional constraint that is specifically violated by voiced geminates in the presence of another voiced obstruent.

5.2 Cumulativity denied: Lyman's Law

The constraint *2-VOICE is motivated by Lyman's Law; for related analyses, see (Alderete 1997, Ito and Mester 2003, Suzuki 1998). Lyman's Law has a "counting-like" feel to it: one voiced obstruent is ok, but 2 are bad. The constraint *VOICE-OBS, violated by voiced obstruents in general, is certainly independently motivated. This sounds like just the sort of "counting threshold" that is seen with cumulative interactions. Why not capture Lyman's Law in this way?

The answer is that it won't work with IDENT-VOICE and *VOICE-OBS. This is illustrated in Table 20. The problem is that the violation differences stay proportional. The two constraints consistently trade off one-to-one: no assignment of weights will yield a cumulative interaction. In this case, such a cumulative interaction is desirable, but unobtainable.

Table 20 Accumulation of *VOI-OBS against IDENT-VOICE won't capture the 1 vs. 2 counting of Lyman's Law.

	IDENT-VOICE	*VOICE-OBS
/pogu/		
poku	1	
pogu		1
/bogu/		
boku	1	1
bogu		2

This is interesting because of the clearly conjunctive nature of *2-VOICE.[10] The constraint is violated precisely when two or more violations of *VOICE-OBS occur in the domain of a word. As PBP:10 point out, a conjoined constraint of this sort can accomplish the 1 vs. 2 counting-like effect in either OT or HG.

This example again echoes one of the main points of PBP: distinguishing differing numbers of violations of a constraint does not come automatically in HG. Not everything that appears “counting-like” is easily accounted for via cumulative interactions. In this case, *VOICE-OBS has proportional violation differences with IDENT-VOICE, while *2-VOICE does not.

5.3 Shifting interactions

Other issues can be observed with longer forms. This comes up in the fourth form listed in (14), /deibiddo/ → [deibitto]. The key issue is the behavior of *2-VOICE when there are more than two voiced obstruents present in the domain (a word).[11] One version would be to have a constraint that combinatorially assesses a separate violation for every distinct pair of distinct violations of *VOICE-OBS. I will call such a constraint *2-VOICE-MULT. Such a constraint continues to accumulate violations in forms with larger numbers of voiced obstruents. It captures the attested behavior for /deibiddo/ → [deibitto]; this is shown in Table 21.

Table 21 The geminate is devoiced: *2-VOICE-MULT accumulates violations combinatorially.

Weight	1.5	1	1	*H*
	IDENT-VOICE	*VCE-GEM	*2-VOICE-MULT	
/deibiddo/				
deibiddo		1	3	-4
☞ deibitto	1		1	-2.5
deipitto	2			-3
deipiddo	1	1	1	-3.5

[10] PBP state that their analysis is inspired by Nishimura’s application of local conjunction theory (Smolensky 1993, Smolensky 2006).
[11] The precise behavior of *2-VOICE for /deibiddo/ candidates isn’t specified in PBP.

The combinatorial growth of the number of distinct pairs of obstruents, assessed by *2-VOICE-MULT, quickly outpaces the linear growth of the number of obstruents, assessed by IDENT-VOICE. This differential leads, not surprisingly, to additional disproportional violation differences. This can be illustrated with a hypothetical input, /deibido/, similar to the previous form but replacing the voiced geminate with a non-geminate voiced obstruent. The result is a three-way tie in which each co-optimum devoices exactly one of the obstruents. This is shown in Table 22.

Table 22 Disproportional violation differences leads to non-geminate devoicing.

	Weight	1.5	1	1	*H*
		IDENT-VOICE	*VCE-GEM	*2-VOICE-MULT	
	/deibido/				
	teipito	3			-4.5
	deipito	2			-3
☞	deibito	1		1	-2.5
☞	deipido	1		1	-2.5
☞	teibido	1		1	-2.5
	deibido			3	-3

What is interesting here is the shifting interactions between IDENT-VOICE and *2-VOICE-MULT. For /webbu/, *2-VOICE-MULT is indifferent as IDENT-VOICE gains a violation for [weppu]: the constraints don't conflict.[12] For /doggu/, *2-VOICE-MULT and IDENT-VOICE evenly trade violations in [doggu] vs. [dokku]: it is *VCE-GEM that pushes the violation difference ratio between the gang and IDENT-VOICE to 2/1. For /deibido/, the [deibito]~[deibido] comparison yields a violation difference ratio between IDENT-VOICE and *2-VOICE-MULT of 2/1 (with *VCE-GEM silent). The same violation difference ratio, 2/1, that allows the (*VCE-GEM, *2-VOICE-MULT) gang to overtake IDENT-VOICE for /doggu/ allows *2-VOICE-MULT to single-handedly overtake IDENT-VOICE for /deibido/.

An alternative version of a constraint like *2-VOICE would simply assess a **single** violation to forms with two or more voiced obstruents in them. I will call such a constraint *2-VOICE-SING. Table 23 shows the /deibiddo/ competition using this constraint. This analysis predicts the wrong output for /deibiddo/, with a geminate: no geminate devoicing is predicted.

[12] One could push the matter, and claim a violation difference ratio of 0/2 (*2-VOICE-MULT over IDENT-VOICE), with the 0 indicating the lack of conflict between the constraints. The choice of which is numerator and which is denominator depends on how one manipulates the inequalities, revealing a quirk in viewing such non-conflicting relationships as ratios: the possibility of zero in the denominator.

Table 23 A non-accumulating version makes a wrong prediction: the geminate is not devoiced.

	Weight	1.5	1	1	*H*
		IDENT-VOICE	*VCE-GEM	*2-VOICE-SING	
	/deibiddo/				
☞	deibiddo		1	1	-2
	deibitto	1		1	-2.5
	deipitto	2			-3
	deipiddo	1	1	1	-3.5

The constraint *2-VOICE-SING distinguishes between one and more than one: its capacity for violation within a word is exhausted once more than one voiced obstruent is present. Thus, devoicing the geminate incurs a violation of IDENT-VOICE, but does not shed a violation of *2-VOICE-SING in exchange: deibitto still has two voiced obstruents, and thus still violates *2-VOICE-SING.

What we have here is a failure to accumulate. In the comparison of [doggu] to [dokku] in Table 19, the violation difference ratio was 2/1: *VCE-GEM and *2-VOICE-SING trading off against IDENT-VOICE. This allowed it to be distinguished from the 1/1 ratios for [bobu]~[bopu] and [webbu]~[weppu]. In the comparison of [deibiddo] to [deibitto] in Table 23, the ratio is 1/1: *VCE-GEM trading off against IDENT-VOICE, the same trade-off as for [webbu]~[weppu]. Because the violation differences are proportional for [deibiddo]~[deibitto] and [webbu]~[weppu], HG cannot distinguish them. Within each of those two comparisons, *2-VOICE-SING does not distinguish the candidates, and thus does not conflict with any other constraints.

Another way to try to "keep accumulating", in an effort to maintain proportionality between the violation differences, would be with an OCP-inspired variant, OCP-VOICEDOBS, that is violated by any pair of voiced obstruents with no intervening obstruent. I must emphasize up front that the only purpose of this is to demonstrate some interesting possible consequences of altering violation accumulation patterns. Relating Lyman's Law to the OCP is neither original nor unproblematic. Within OT, the OCP is invoked as inspiration for constraint conjunction approaches to Lyman's Law and to dissimilatory processes in general by Alderete (1997); see Ito and Mester (2003) for a recent discussion of the issues.[13]

In the shorter forms, OCP-VOICEDOBS is violated in the same circumstances as *2-VOICE-MULT / *2-VOICE-SING, and thus is able to gang up with *VCE-GEM against IDENT-VOICE. With OCP-VOICEDOBS, OT still cannot simultaneously satisfy [bobu]≻[bopu], [webbu]≻[weppu], and [dokku]≻[doggu]. HG can: with the same weights, OCP-VOICEDOBS is able to gang up with *VCE-GEM against IDENT-VOICE for [dokku]≻[doggu].

The resulting tableau for /deibiddo/ is shown in Table 24. The two cells where the violation count of OCP-VOICEDOBS differs from that of *2-VOICE-SING are shaded.

[13] The possibility of an OCP-like version of *2-VOICE is also noted in PBP:13.

Table 24 Two devoicing options tie.

	Weight	1.5	1	1	*H*
		IDENT-VOICE	*VCE-GEM	OCP-VOICEDOBS	
	/deibiddo/				
	deibiddo		1	2	-3
☞	deibitto	1		1	-2.5
	deipitto	2			-3
☞	deipiddo	1	1		-2.5

Unlike *2-VOICE-SING, OCP-VOICEOBS only assesses two violations to [deibiddo], re-establishing the proportional violation pattern of 2/1 (OCP-VOICEDOBS and *VCE-GEM over IDENT-VOICE) for [deibiddo]~[deibitto], once again matching the trade-off for [doggu]~[dokku]. For unbroken strings of voiced obstruents, the violations of OCP-VOICEOBS will accumulate linearly, in contrast to *2-VOICE-MULT. The indicated constraint weights achieve one of the desired comparisons: [deibitto]≻[deibiddo].

But the change in constraint also removes a violation from [deipiddo]. This introduces a preference in the comparison [deibiddo]~[deipiddo], where *2-VOICE-SING was indifferent. The substitution of OCP-VOICEDOBS for *2-VOICE-SING rescues [deipiddo] from simple harmonic bounding obscurity, and reverses another comparison: now, [deipiddo]≻[deibiddo].[14]

Furthermore, a civil war has broken out: in the comparison between [deibitto] and [deipiddo], the members of the original gang, OCP-VOICEDOBS and *VCE-GEM, have turned on each other, and directly conflict. The limit of treating OCP-VOICEDOBS and *VCE-GEM as undifferentiated members of a gang has been reached. The two function as a gang on the shorter forms, but not here.

Because the same weight was assigned to both of the lower weighted constraints, the result is a tie for optimality between [deibitto] (avoiding a voiced geminate) and [deipiddo] (avoiding an OCP violation). Note that this is an instance of candidates with non-identical violation profiles tying for optimality (section 2.4).

The tie can be broken by distinguishing the weights for *VCE-GEM and OCP-VOICEDOBS. HG permits them to be distinguished in either direction. A weighting with w(*VCE-GEM) > w(OCP-VOICEDOBS) will make [deibitto], with the devoiced geminate, optimal. A weighting with w(OCP-VOICEDOBS) > w(*VCE-GEM) will instead make [deipiddo], with the obstruent voicing contour, optimal. The latter case is shown in Table 25.

[14] The linear violation accumulation might be obtained without this latter consequence by building the counting threshold of 2 directly into a constraint: a violation assessed for each voiced obstruent beyond one. I won't pursue this possbility further here.

Table 25 The tie is broken by distinguishing the weights of *VCE-GEM and OCP-VOICEDOBS.

Weight	1.5	1	1.1	*H*
	IDENT-VOICE	*VCE-GEM	OCP-VOICEDOBS	
/deibiddo/				
deibiddo		1	2	-3.2
deibitto	1		1	-2.6
deipitto	2			-3
☞ deipiddo	1	1		-2.5

As described above, the change to OCP-VOICEDOBS from *2-VOICE-SING establishes the violation difference ratio of 2/1 (OCP-VOICEDOBS and *VCE-GEM over IDENT-VOICE) for [deibiddo]~[deibitto], once again matching the ratio for [doggu]~[dokku]. The same constraint change also establishes a ratio of 2/1 for [deibiddo]~[deipiddo], which is disproportionate when compared with the 1/1 ratio for [bobu]~[bopu]. With *2-VOICE-SING, the constraints don't conflict on [deibiddo]~[deipiddo] (hence the harmonic bounding of [deipiddo]). The constraint change allows [deipiddo]≻[deibiddo] to be satisfied, and further to be satisfied simultaneously with [bobu]≻[bopu], as a consequence of the disproportional violation differences. Eliminating one cumulative interaction has introduced a different one.[15]

Again, the point of this exercise is not to argue for or against a particular analysis of the Japanese loanword data, or of Lyman's Law; more sophisticated discussion of the empirical phenomena is to be found elsewhere (Ito and Mester 2003, Kawahara 2006, Nishimura 2003).[16] The constraints *2-VOICE-SING and OCP-VOICEDOBS are constructions to illustrate how interactions can shift across comparisons. Changing a constraint to maintain proportional violation differences between some candidate comparisons can create disproportional violation differences for other comparisons. Constraints which gang up in one comparison may conflict on another.

6 Bounded Violation Counts

HG systems can yield infinite typologies (typologies with an infinite number of distinct languages), as has been demonstrated by Legendre et al. (2006). An example of this exists in the "minimum coda cluster" phenomenon described in section 3.2: there are an infinite number of grammars, including at least one for each minimum coda cluster size. Another example, presented by Legendre et al. (and discussed in PBP) is the possibility of stress windows of arbitrary size. In PBP it is observed that making an infinite number of distinctions in the manner of these examples requires that at least one constraint be capable of assigning an unbounded number of violations. The point is driven home with a proof that if all constraints have a fixed bound on the number of violations they can assign, then an HG system over those constraints defines a finite typology.

PBP ultimately advocate Local Harmonic Serialism as a way to prevent many of the undesirable patterns they discuss, and do not comment on the issue of finite vs. infinite typologies with respect to Local

[15] This moral can be spun in more than one way. An alternative narrative is to say that the change in constraint established disproportional violation differences between [deibiddo]~[deibitto] and ([bobu]~[bopu] and [webbu]~[weppu]), with the unintended consequence of also establishing a disproportional violation difference between [deibiddo]~[deipiddo] and ([bobu]~[bopu] and [webbu]~[weppu]). One cumulative interaction begets another.

[16] See also PBP for discussion of both the phenomena and of learning and the incorporation of Japanese loan words.

Harmonic Serialism. In adopting Local Harmonic Serialism, they forsake extrinsic bounds on constraint violation counts in favor of effects of the evaluation mechanism in Local Harmonic Serialism. I won't comment further Local Harmonic Serialism here; see PBP for details. But the prospect of bounding constraint violation counts has precedent in previous work, notably in the work of Frank & Satta (1998), in the context of generative complexity concerns and finite-state grammar models of OT. It is worth contemplating the significance of the observation about bounded violation counts within HG using global optimization.

First, it is worth asking: to what extent is the possibility of an infinite typology really a concern? In linguistics, it is common to observe a pattern that repeats over increasingly larger forms within a single language, and conclude that the language is infinite if there is no obvious motivation for a limit to the pattern. It seems reasonable to adopt the same stance towards typology. If the empirical observation is that there are no "arbitrary counting" patterns in the world's languages, then the prediction by a theory of such patterns themselves would seem to be a larger concern than the prediction that there are an infinite number of variations. If a theory predicts languages with a minimum coda cluster, that seems like a more basic concern than the prediction that an unbounded number of variations of the pattern exist.[17]

Simply bounding the number of violations that can be assessed by the constraints giving rise to an undesirable pattern does not eliminate the undesirable pattern, it simply limits the number of variations of that pattern. Limiting the number of violations of MAX may put a bound on the largest minimum coda cluster size, but you still have languages with minimum coda clusters.[18] If a typology predicts stress windows of two syllables, three syllables, four syllables, five syllables, six syllables, and so forth but only up to 20 syllables, it isn't clear that much comfort should be derived from the 20 syllable bound. Bounding constraint violation counts will limit the number of variations, but isn't an adequate solution to the undesirable interactions.

If one chooses the goal of avoiding cumulative interactions, the key is to maintain proportional violation differences. If a ratio between violation differences is too large, and needs to be reduced to maintain proportionality with another, there are two ways to go about it: by decreasing the numerator, or by **increasing** the denominator. In fact, two of the attempts to maintain proportionality examined in this paper actually increased the number of violations assessed by a constraint to candidates. Changing NOCODA to assess a violation to every segment in a coda position (section 3.3) increases the number of violations assessed to many candidates, a change that seems in the opposite direction from imposing a fixed bound on the number of violations. The same applies to the attempt to extend the Japanese loanword analysis by changing *2-VOICE-SING to a constraint which can distinguish numbers of voiced obstruents beyond 2 (section 5.3): the effect is an increase in the number of violations for many candidates.

In fact, *2-VOICE-SING is the only example of a constraint discussed here which actually has a finite bound on the number of violations (within the domain of a word, at least), and that very property led to problematic cumulative interactions. Any constraint which independently evaluates structures that can occur repeatedly in forms of increasing length can be expected to have unbounded violation potential. The cases actually demonstrated here where cumulative interactions are avoided do not involve bounded violation counts: they maintain proportional violation differences across all competitions.

[17] It bears emphasizing that PBP make no claim to the contrary; they advocate using Local Harmonic Serialism to prevent even finitely-restricted versions of patterns like minimum coda clusters.
[18] Along with relatively odd predictions for inputs where the number of potential coda consonants exceeds the allotted number of violations of MAX.

7 Conclusions

The differences between lexicographic ordering and linear numeric ordering have the consequence that OT and HG doubly diverge. Neither is a special case of the other, as each can describe patterns that the other cannot. Many cases permitting OT and HG to diverge can be understood in terms of disproportional violation differences. This includes, within limits, gang effects: this paper has applied violation difference ratios to cases where the members of an identified gang do not conflict on any of the relevant comparisons. This was demonstrated in sections 3 and 4. Patterns of cumulative interaction can be complex, as is suggested by the example in section 5.

Cumulative interactions are avoided when constraints that conflict on a variety of comparisons always maintain proportionality in the differences in the number of violations across the comparisons. This can be maintained when the conflicting constraints accumulate violations in proportional amounts. The potential for cumulative interactions arises when proportional accumulation is not maintained. Such lack of maintenance can be as easily attributed to one constraint failing to accumulate a sufficient number of violations as to the other constraint accumulating too many. Adhering to a constant ratio of violation differences, regardless of the magnitudes of the violation differences, can prevent the possibility of cumulative interactions, be they desirable or undesirable.

Bounding the violation counts of all constraints prevents the realization of arbitrarily skewed trade-offs of violations in comparisons. But that does not prevent trade-offs from differing disproportionately; it just limits the variety of realizable disproportions. Bounding the violation counts of all constraints guarantees that an HG analysis exists which matches any OT one for that system, but does not typologically prevent non-OT cumulative interactions.

Violation difference ratios do not provide a complete general case analysis of the possible divergences between lexicographic and linear numeric ordering, but they make a number of cases easy to understand. Such analysis reveals cases that are not “Anything Goes” situations, but nevertheless do not permit cumulative interactions. Even when the violation difference ratios are not equivalent to 1/1, cumulative interactions are prevented if the violation difference ratios remain constant across comparisons.

8 References

Alderete, John. 1997. Dissimilation as local conjunction. In *Proceedings of the Twenty-Seventh Meeting of the NorthEast Linguistics Society*, ed. by Kiyomi Kusumoto, 17-32. Amherst, MA: GLSA.

Frank, Robert, and Satta, Giorgio. 1998. Optimality Theory and the generative complexity of constraint violability. *Computational Linguistics* 24:307-315.

Ito, Junko, and Mester, Armin. 2003. *Japanese Morphophonemics: Markedness and Word Structure*: MIT Press Linguistic Inquiry Monograph Series 41. Cambridge, MA.

Johnson, Mark. 2002. Optimality-theoretic Lexical Functional Grammar. In *The Lexical Basis of Sentence Processing: Formal, Computational and Experimental Issues*, ed. by Paula Merlo and Suzanne Stevenson, 59-74. Amsterdam, The Netherlands: John Benjamins.

Kawahara, Shigeto. 2006. A faithfulness ranking projected from a perceptibility scale: The case of [+voice] in Japanese. *Language* 82:536-574.

Keller, Frank. 2000. Gradience in grammar: Experimental and computational aspects of degrees of grammaticality. PhD. dissertation, University of Edinburgh.

Keller, Frank, and Asudeh, Ash. 2002. Probabilistic learning algorithms and Optimality Theory. *Linguistic Inquiry* 33:225-244.

Keller, Frank. 2006. Linear Optimality Theory as a model of gradience in grammar. In *Gradience in Grammar: Generative Perspectives*, ed. by G. Fanselow, C. Fery, R. Vogel, and M. Schlesewsky. Oxford: Oxford University Press.

Legendre, Geraldine, Miyata, Yoshiro, and Smolensky, Paul. 1990a. Harmonic Grammar - a formal multi-level connectionist theory of linguistic well-formedness: An application. In *Proceedings of the Twelfth Annual Meeting of the Cognitive Science Society*, 884-891.

Legendre, Geraldine, Miyata, Yoshiro, and Smolensky, Paul. 1990b. Harmonic Grammar - a formal multi-level connectionist theory of linguistic well-formedness: Theoretical foundations. In *Proceedings of the Twelfth Annual Meeting of the Cognitive Science Society*, 388-395.

Legendre, Geraldine, Sorace, Antonella, and Smolensky, Paul. 2006. The Optimality Theory - Harmonic Grammar connection. In *The Harmonic Mind*, ed. by Paul Smolensky and Geraldine Legendre, 339-402. Cambridge, MA: MIT Press.

Lombardi, Linda. 2001. Why Place and Voice are Different: Constraint-Specific Alternations in Optimality Theory. In *Segmental Phonology in Optimality Theory: Constraints and Representations*, ed. by Linda Lombardi, 13-45. Cambridge University Press.

McCarthy, John. 2006a. Slouching toward optimality: Coda-reduction in OT-CC. Ms. ROA-878.

McCarthy, John J. 2006b. Restraint of analysis. In *Wondering at the Natural Fecundity of Things: Essays in Honor of Alan Prince*, ed. by Eric Bakovic, Junko Ito, and John J. McCarthy. ROA-844.8.

McCarthy, John J. 2007. *Hidden Generalizations: Phonological Opacity in Optimality Theory*: Advances in Optimality Theory. Oakville, CT: Equinox.

Nishimura, K. 2003. Lyman's law in loanwords. Ms., University of Tokyo.

Pater, Joe. 2007. The power of weighted constraints. Handout of talk given at the Johns Hopkins University, Feb. 8.

Pater, Joe, Bhatt, Rajesh, and Potts, Christopher. 2007a. Linguistic Optimization. Ms., Linguistics Department, University of Massachusetts, Amherst. ROA-924.

Pater, Joe, Potts, Christopher, and Bhatt, Rajesh. 2007b. Harmonic Grammar with linear programming. Ms., University of Massachusetts, Amherst. ROA-872.

Prince, Alan, and Smolensky, Paul. 1993/2004. *Optimality Theory: Constraint interaction in generative grammar*. Malden, MA: Blackwell.

Prince, Alan. 2002. Anything goes. In *New Century of Phonology and Phonological Theory*, ed. by T. Honma, M. Okazaki, T. Tabata, and S. ichi Tanaka, 66-90. Tokyo: Kaitakusha. ROA-536.

Samek-Lodovici, Vieri, and Prince, Alan. 1999. Optima. Technical Report RuCCS-TR-57, Center for Cognitive Science, Rutgers University, New Brunswick, NJ. ROA-363.

Smolensky, Paul. 1993. Harmony, markedness, and phonological activity. Handout, Rutgers Optimality Workshop -1. ROA-87.

Smolensky, Paul. 2006. Optimality in phonology II: Harmonic completeness, local constraint conjunction, and feature domain markedness. In *The Harmonic Mind*, ed. by Paul Smolensky and Geraldine Legendre, 27-160. Cambridge, MA: MIT Press.

Suzuki, K. 1998. A Typological Investigation of Dissimilation. PhD. dissertation, University of Arizona, Tucson, AZ.