Towards a Factorization of String-Based Phonology

Abstract. Inspired by the model-theoretic treatment of phonology in Potts & Pullum (2002) and Kracht (2003), we develop an extendable modal logic for the investigation of string-based phonology. In contrast to previous research in this vein (Russell 1993, Kaplan & Kay 1994, Mohri & Sproat 1996), we ultimately strive to study the entire class of such theories rather than merely one particular incarnation thereof. To this end, we first provide a formalization of classic Government Phonology in a restricted variant of temporal logic, whose generative capacity is then subsequently increased by the addition of further operators, moving us along the subregular hierarchy until we reach the regular stringsets. We then identify several other axes along which Government Phonology might be generalized, moving us towards a parametric metatheory of phonology.

Like any other subfield of linguistics, phonology is home to a multitude of competing theories that differ vastly in their conceptual and technical assumptions. Contentious issues are, among others, the relation between phonology and phonetics (and if it is an interesting research question to begin with), if features are privative, binary or attribute valued, if phonological structures are strings or trees, if features can move from one position to another (i.e. if they are *autosegments*), and what role optimality requirements play in determining well-formedness. Meticulous empirical comparisons carried out by linguists have so far failed to yield conclusive results; it seems that for every phenomenon that lends support to certain assumptions, there is another one that refutes them. We do not think that this constitutes a problem to phonological research. Unless we assume that scientific theories can indeed reflect reality as it is rather than merely approximate it, it is to be expected that one theory may fail where another one succeeds and *vice versa*. A similar situation arises in physics, where depending on the circumstances light is thought to exhibit particle-like or wave-like properties.

But given this apparent indeterminacy of theory choice, it is only natural to ask if we can identify classes of interchangeable theories, i.e. proposals which look different superficially but are the same in any other respect. On a bigger scale, this requires developing a metatheory of phonology that uses a finite set of parameters to conclusively determine the equivalence class to which a given phonological theory belongs. This paper aims to lay the basis for such a metatheory using techniques originating in model-theoretic syntax (Blackburn & Meyer-Viol 1994, Kracht 1995, Rogers 2003). We feel obliged to point out in advance that we doubt that a linguistically adequate formal theory of phonology is attainable. However, we also think that in attempting to construct such a metatheory, one gains crucial insights into the core claims about language that are embodied by different phonological assumptions (e.g. computational complexity and memory usage) and how one may translate those claims from one theory into another. Moreover, the explicit logical formalization of linguistic theories allows us to investigate various problems in an algorithmic way using techniques from proof theory and model checking. These insights are relevant to linguists and computer scientists alike. Linguists get a better understanding of how their claims relate to the psychological reality of language, how the different modules of a given theory interact to yield generalizations and how they increase the expressivity of a theory (see Potts & Pullum (2002) for such results on optimality theory). To a limited degree, they also get the freedom to switch to a different theory for specific phenomena without jeopardizing the validity of their framework of choice. Computer scientists, on the other hand, will find that the model-theoretic perspective on phonology eases the computational implementation of linguistic proposals and allows them to gauge their runtime-behavior in advance. Furthermore, they may use the connection between finite model theory and formal language theory to increase the efficiency of their programs by picking the weakest phonological theory that is expressive enough for the task at hand.

This paper is divided into two parts as follows. First, we introduce Government Phonology as an example of a weak theory of phonology and show how it can be axiomatized as a theory of richly annotated string structures using modal logic. In the second part, we analyze several parameters that might have an effect on the generative capacity of our formalization of GP. We show that increasing the power of the spreading operation moves us along the subregular hierarchy and that different types of feature systems have no effect on expressivity in general. We close with a short discussion of two important areas of future research, the impact of the syllable template on generative capacity and the relation between derivational and representational theories.

The reader is expected to have some basic familiarity with formal language theory, non-classical logics and model-theoretic syntax. There is an abundance of introductory material for the former two, while the latter is cogently summarized in Rogers (1996) and Pullum (2007).

1. A Weak Theory of Phonology — Government Phonology

1.1. Informal Overview

Due to space restrictions, we offer but a sketch of the main ideas of Government Phonology (GP), and the reader is advised to check the exposition against the examples in figure 1 on the facing page. First, though, a note on our sources is in order. Just like Government-and-Binding theory, GP has changed a lot since its inception and practitioners hardly ever fully specify the details of the version of GP they use. However, there seems to be a consensus that a GP-variant is considered canonical if it incorporates the following modules: government, the syllable template, coda licensing and the ECP from Kaye, et al. (1990), magic licensing from Kaye (1992), and licensing constraints and the revised theory of elements from Kaye (2000). Our general strategy is to follow the definitions in Kaye (2000) as closely as possible and fill in any gaps using the relevant literature. The interested reader might also want to consult Graf (2009) for an in-depth discussion of GP.

In GP, the carrier of all phonological structure is the *skeleton*, a finite, linearly ordered sequence of nodes to which phonological expressions (PEs) can be attached in order to form the *melody* of the structure. A PE is built from a set E of privative features called *elements*, yielding a pair $\langle O, H \rangle$, $O \subseteq E$ a set of *operators*, $H \in E \cup \{\emptyset\}$ the *head*, and $H \notin O$. It is an open empirical question how many features are needed for an adequate account of phonological behavior (Jensen 1994, Harris & Lindsey

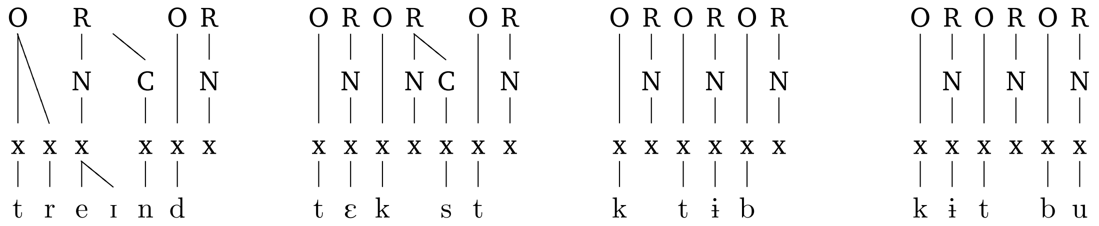

Figure 1: Some phonological structures in GP (with IPA notation)

1995) — Kaye (2000) fixes $E := \{\mathrm{A}, \mathrm{I}, \mathrm{U}, \mathrm{H}, \mathrm{L}, \text{ʔ}\}$, but for our axiomatization the only requirement is for E to be finite. The set of licit PEs is further restricted by language-specific *licensing constraints*, i.e. restrictions on the coocurrence of features and their position in the PE. Some examples of PEs are [s] $= \langle \{A, H\}, \emptyset \rangle$, [n] $= \langle \{L, \text{ʔ}\}, A \rangle$, [ɨ] $= \langle \emptyset, \emptyset \rangle$, [ɪ] $= \langle \{I\}, \emptyset \rangle$, [i] $= \langle \emptyset, I \rangle$, and [j] $= \langle \emptyset, I \rangle$.

As the last two examples show, every PE is inherently underspecified; whether it is realized as a consonant or a vowel depends on its position in the structure, which is annotated with constituency information. An expression is realized as a vowel if it is associated to a node contained by a *nucleus* (N), but as a consonant if the node is contained by an *onset* (O) or a *coda* (C). Every N constitutes a *rhyme* (R), with C an optional subconstituent of R. All O, N and R may *branch*, that is be associated to up to two nodes (by transitivity of containment, a branching R cannot contain a branching N). Furthermore, word initial O can be *floated*, i.e. be associated to no node at all. The number of PEs per node is limited to one, with the exception of unary branching N, where the limit is two (to model light diphthongs).

All phonological structures are obtained from concatenating $\langle \mathrm{O}, \mathrm{R} \rangle$ pairs according to constraints imposed by two government relations. *Constituent government* restricts the distribution of elements within a constituent, requiring that the leftmost PE licenses all other constituent-internal PEs. *Transconstituent government* enforces dependencies between the constituents themselves. In particular, every branching O has to be licensed by the N immediately following it, and every C has to be licensed by the PE contained in the immediately following O. Even though the precise licensing conditions are not fully worked out for either government relation, the general hypothesis is that PE_i licenses PE_j iff PE_i is leftmost and contained by N, or leftmost and composed from at most as many elements as PE_j and licenses no $PE_k \neq PE_j$ (hence any C has to be followed by a non-branching O, but a branching O might be followed by a branching N or R).

GP also features *empty categories*: a non-coda segment associated solely to the PE $\langle \emptyset, \emptyset \rangle$ can optionally remain unpronounced. For O, this is lexically specified. For N, on the other hand, it is determined by the phonological ECP, which allows only non-branching *p-licensed* N to be mapped to the empty string. N is licensed if it is followed by a coda containing a sibilant (*magic licensing*), or in certain languages if it is the rightmost segment of the string (*final empty nucleus*, abbreviated FEN), or if it is *properly governed* (Kaye 1990). N is properly governed if the first N following it is not p-licensed and no government relations hold between or within any Cs or Os in-between the two Ns.

Finally, GP allows elements to *spread*, just as in fully autosegmental theories (Goldsmith 1976). All elements, though, are assumed to share a single tier, and association lines are allowed to cross. The properties of spreading have not been explicitly spelled out in the literature, but it is safe to assume that it can proceed in either direction and might be optional or obligatory, depending on the element, its position in the string and the language in question. While there seem to be restrictions on the set of viable targets given a specific source, the only canonical one is a ban against spreading within a branching O.

1.2. Formalization in Modal Logic

For our formalization, we use a very weak modal logic that can be thought of as the result of removing the "sometime in the future" and "sometime in the past" modalities from restricted temporal logic (Cohen, et al. 1993, Etessami, et al. 1997).

Let E be some non-empty finite set of *basic elements* different from the neutral element v, which represents the empty set of GP's feature calculus. We define the set of *elements* $\mathscr{E} := (E \times \{1,2\} \times \{head, onset\} \times \{local, spread\}) \cup (\{v\} \times \{1,2\} \times \{head, onset\} \times \{local\})$. The set of melodic features $\mathscr{M} := \mathscr{E} \cup \{\mu, \texttt{fake}, \checkmark\}$ will be our set of propositional variables. We employ μ (mnemonic for *mute*) and $\checkmark$ to mark unpronounced and licensed segments, respectively, and `fake` for unassociated onsets. For the sake of increased readability, the set of propositional variables is "sorted" such that $x \in \mathscr{M}$ is represented by m, $m \in \mathscr{E}$ by e, heads by h, operators by o. The variable e_n is taken to stand for any element such that $\pi_2(e) = n$, where $\pi_i(x)$ returns the i^{th} projection of x. In rare occasions, we will write $\underline{e}$ and $\overline{e}$ for a specific element e in head and operator position, respectively.

We furthermore use three nullary modalities[1], N, O, C, the set of which we designate by $\mathscr{S}$, read *skeleton*. In addition, we have two unary diamond operators $\lhd$ and $\rhd$, whose respective duals are denoted by $\blacktriangleleft$ and $\blacktriangleright$. The set of well-formed formulas is built up in the usual way from $\mathscr{M}$, $\mathscr{S}$, $\lhd$, $\rhd$, $\rightarrow$ and $\bot$.

Our models $\mathfrak{M} := \langle \mathfrak{F}, V \rangle$ are built over bidirectional frames $\mathfrak{F} := \langle D, R_i, R_\lhd \rangle_{i \in \mathscr{S}}$, where $D \subseteq \mathbb{N}$, and $R_i \subseteq D$ for each $i \in \mathscr{S}$, and $R_\lhd$ is the successor function over $\mathbb{N}$. The valuation function $V : \mathscr{M} \rightarrow \wp(D)$ maps propositional variables to subsets of D. The definition of satisfaction is standard.

$$
\begin{array}{lll}
\mathfrak{M}, w \models \bot & & \text{never} \\
\mathfrak{M}, w \models p & \text{iff} & w \in V(p) \\
\mathfrak{M}, w \models \neg\phi & \text{iff} & \mathfrak{M}, w \not\models \phi \\
\mathfrak{M}, w \models \phi \wedge \psi & \text{iff} & \mathfrak{M}, w \models \phi \text{ and } \mathfrak{M}, w \models \psi \\
\mathfrak{M}, w \models N & \text{iff} & w \in R_N \\
\mathfrak{M}, w \models O & \text{iff} & w \in R_O \\
\mathfrak{M}, w \models C & \text{iff} & w \in R_C \\
\mathfrak{M}, w \models \lhd \phi & \text{iff} & \mathfrak{M}, w+1 \models \phi \\
\mathfrak{M}, w \models \rhd \phi & \text{iff} & \mathfrak{M}, w-1 \models \phi
\end{array}
$$

The formalization of the skeleton is straightforward if we model binary branching constituents as two adjacent unary branching ones and view rhymes as mere nota-

[1]We follow the terminology of Blackburn, et al. (2002) here. Nullary modalities correspond to unary relations and can hence be thought of as propositional constants.

tional devices. Observe that we implement Ns containing diphthongs as single N with both e_1 and e_2 elements associated to it.

S1	$\bigwedge_{i\in\mathscr{S}}(i \leftrightarrow \bigwedge_{i\neq j\in\mathscr{S}} \neg j)$	Unique constituency
S2	$(\blacktriangleleft \bot \rightarrow O) \wedge (\blacktriangleright \bot \rightarrow N)$	Word edges
S3	$R \leftrightarrow (N \vee C)$	Definition of rhyme
S4	$N \rightarrow \lhd O \vee \lhd N$	Nucleus placement
S5	$O \rightarrow \neg \lhd O \vee \neg \rhd O$	Binary branching onsets
S6	$R \rightarrow \neg \lhd R \vee \neg \rhd R$	Binary branching rhymes
S7	$C \rightarrow \lhd N \wedge \rhd O$	Coda placement

GP's feature calculus is also easy to capture. A propositional formula ϕ over a set of variables $x_1, \ldots, x_k$ is called *exhaustive* iff $\phi := \bigwedge_{1\leq i\leq k} \psi_i$, where for every i, ψ_i is either x_i or $\neg x_i$. A PE ϕ is an exhaustive propositional formula over $\mathscr{E}$ such that $\phi \cup \{\mathbf{F1}, \mathbf{F2}, \mathbf{F3}, \mathbf{F4}, \bigvee h\}$ is consistent.

F1	$\bigwedge(h_n \rightarrow \bigwedge_{h_n\neq h'_n} \neg h'_n)$	Exactly one head
F2	$\neg \underline{v} \rightarrow \bigwedge(h_n \rightarrow \bigwedge_{\pi_1(h)=\pi_1(o)} \neg o_n)$	No basic element (except v) twice
F3	$\overline{v} \rightarrow \bigwedge_{o\neq\overline{v}} \neg o$	$\overline{v}$ excludes other operators
F4	$\bigwedge(e_2 \rightarrow \bigvee h_1 \wedge \bigvee o_1)$	Pseudo branching implies first branch

Let *PH* be the least set containing all such ϕ, and let $lic : PH \rightarrow \wp(PH)$ map every ϕ to its set of melodic licensors. By $S \subseteq PH$ we designate the set of PEs occurring in magic licensing configurations (the letter S is mnemonic for "sibilants"). The following five axioms, then, sufficiently restrict the melody.

M1	$\bigwedge_{i\in\mathscr{S}} (i \rightarrow (\bigvee h_1 \wedge \bigvee o_1) \vee \mu \vee \texttt{fake})$	Universal annotation
M2	$((O\vee \lhd N \vee \rhd N) \rightarrow \bigwedge \neg e_2)$	No pseudo branching for O, C & branching N
M3	$O\wedge \lhd O \rightarrow \bigwedge_{\phi\in PH}(\phi \rightarrow \bigvee_{\psi\in lic(\phi)} \lhd \psi)$	Licensing within branching onsets
M4	$C \wedge \bigwedge_{i\in S} \neg i \rightarrow \lhd \neg\mu \wedge \bigwedge_{\phi\in PH}(\phi \rightarrow \bigvee_{\psi\in lic(\phi)} \rhd \psi)$	Melodic coda licensing
M5	$\texttt{fake} \rightarrow O \wedge \bigwedge_{m\neq\texttt{fake}} \neg m$	Fake onsets

Remember that GP allows languages to impose further restrictions on the melody by recourse to licensing constraints. It is easy to see that licensing constraints operating on single PEs can be captured by propositional formulas. The licensing constraint "A must be head", for instance, corresponds to the propositional formula $\neg \overline{A}$. Licensing constraints that extend beyond a single segment can be modeled using $\lhd$ and $\rhd$, provided their domain of application is finitely bounded. See Graf (2009) and the discussion on spreading below for further details.

As mentioned above, we use μ to mark "mute" segments that will be realized as the empty string. The distribution of μ is simple for O and C — the former always allows it, the latter never does. For N, we first need to distribute $\checkmark$ in a principled manner across the string to mark the licensed nuclei, which may remain unpronounced. Note that $\underline{v} \wedge \overline{v}$ by itself does not designate unpronounced segments (remember the PE for [ə]), and that unpronounced segments may not contain any other elements (which would affect spreading).

L1	$\mu \rightarrow \neg C \wedge (N \rightarrow \checkmark) \wedge \underline{v} \wedge \overline{v}$	Empty categories
L2	$N \wedge \lhd N \rightarrow (\mu \leftrightarrow \lhd \mu)$	Licensing of branching nuclei
L3	$O \wedge \lhd O \rightarrow \neg \lhd \mu \wedge \neg \mu \wedge \neg \rhd \mu$	Licensing of branching onsets
L4	$N \wedge \checkmark \leftrightarrow \underbrace{\rhd (C \wedge \bigvee_{i \in S} i)}_{\text{Magic Licensing}} \vee \underbrace{(\neg \lhd N \wedge \blacktriangleright \bot)}_{\text{FEN}} \vee \underbrace{((\neg \lhd N \rightarrow \lhd (\lhd N \vee \blacktriangleleft \bot)) \wedge (\neg \rhd N \rightarrow \rhd\rhd (N \wedge \neg \mu)))}_{\text{Proper Government}}$	P-licensing

Axiom **L4** looks daunting at first, but it is easy to unravel. The magic licensing conditions tells us that N is licensed if it is followed by a sibilant in coda position.[2] The FEN condition ensures that wordfinal N are licensed if they are non-branching. The proper government condition is the most complex one, though it is actually simpler than the original GP definition. Remember that N is properly governed if the first N following it is pronounced and neither of the two licenses a branching onset. Also keep in mind that we treat a binary branching constituent as two adjacent unary branching constituents. The proper government condition then enforces a structural requirement such that N (or the first N is we are talking about two adjacent N) may not be preceded by two constituents that are not N and (the second N) may not be followed by two constituents that are not N or not pronounced. Given axioms **S1**–**S7**, this gives the same results as the original constraint.

The last module, spreading, is also the most difficult to accommodate. Most properties of spreading are language specific — only the set of spreadable features and the ban against onset internal spreading are universal. To capture this variability, we define a general spreading scheme σ with six parameters i, j, ω, ϖ, *min* and *max*.

$$\sigma := \bigwedge_{\pi_1(i)=\pi_1(j)} (i \wedge \omega \rightarrow \bigvee_{min \leq n \leq max} \Diamond^n (j \wedge \varpi) \wedge (O \wedge \Diamond O \rightarrow \bigvee_{min+1 \leq n \leq max} \Diamond^n (j \wedge \varpi)))$$

The variables $i, j \in \mathscr{E}$, coupled with judicious use of the formulas ω and ϖ regulate the optionality of spreading. If spreading is optional, i is a spread element and ω, ϖ are formulas describing, respectively, the structural configuration of the target of spreading and the set of licit sources for spreading operations to said target. If spreading is mandatory, then i is a local element and ω, ϖ describe the source and the set of targets. If we want spreading to be mandatory in only those cases where a target is actually available, ω has to contain the subformula $\bigvee_{min \leq n \leq max} \Diamond^n \varpi$. Observe moreover that we need to make sure that every structural configuration is covered by some ω, so that unwanted spreading can be blocked by making ϖ not satisfiable. As further parameters, the finite values $min, max > 0$ encode the minimum and maximum

[2]Note that we can easily restrict the context, if this appears to be necessary for empirical reasons. Strengthening the condition to $\rhd (C \wedge \bigvee_{i \in S} i) \wedge \lhd \blacktriangleleft \bot$, for example, restricts magic licensing to the N occupying the second position in the string.

distance of spreading, respectively. Finally, the operator $\Diamond \in \{\lhd, \rhd\}$ fixes the direction of spreading for the entire formula ($\Diamond^n$ is the n-fold iteration of $\Diamond$). With optional spreading, the direction of the operator is opposite to the direction of spreading, otherwise they are identical.

As the astute reader has probably noticed by now, nothing in our logic prevents us from defining alternative versions of GP. Whether this is a welcome state of affairs is a matter of perspective. On the one hand, the flexibility of our logic ensures its applicability to a wide range of different variants of GP, e.g. to versions where spreading is allowed within onsets or where the details of proper government and the restrictions on branching vary. On the other hand, it begs the question if there isn't an even weaker modal logic that is still expressive enough to formalize GP. The basic feature calculus of GP already requires the logical symbols $\neg$ and $\wedge$, giving us the complete set of logical connectives, and we need $\lhd$ and $\rhd$ to move us along the phonological string. Hence, imposing any further syntactic restrictions on formulas requires advanced technical concepts such as the number of quantifier alternations. However, we doubt that such a move would have interesting ramifications given our goals; we do not strive to find the logic that provides the best fit for a specific theory but to study entire classes of string-based phonological theories from a model-theoretic perspective. In the next section, we try to get closer to this goal.

2. The Parameters of Phonological Theories

2.1. Elaborate Spreading — Increasing the Generative Capacity

It is easy to see that our logic is powerful enough to account for all finitely bounded phonological phenomena (note that this does not imply that GP itself can account for all of them, since certain phenomena might be ruled out by, say, the syllable template or the ECP). In fact, it is even possible to accommodate many long-distance phenomena in a straight-forward way, provided that they can be reinterpreted as arising from iterated application of finitely bounded processes or conditions. Consider for example a stress rule for language L that assigns primary stress to the last syllable that is preceded by an even number of syllables. Assume furthermore that secondary stress in L is trochaic, that is to say it falls on every odd syllable but the last one. Let 1 and 2 stand for primary and secondary stress, respectively. Unstressed syllables are assigned the feature 0. Then the following formula will ensure the correct assignment of primary stress (for the sake of simplicity, we assume that every node in the string represents a syllable; it is an easy but unenlightening exercise to rewrite the formula for our GP syllable template).

$$\bigvee_{i \in \{0,1,2\}} i \wedge \bigwedge_{i \neq j \in \{0,1,2\}} (i \rightarrow \neg j) \wedge (\blacktriangleleft \bot \rightarrow 1 \vee 2) \wedge (2 \rightarrow \rhd\, 0) \wedge$$
$$(0 \rightarrow \rhd\, (1 \vee 2) \vee \blacktriangleright \bot) \wedge (1 \rightarrow \neg \lhd 1 \wedge (\blacktriangleright \bot \vee \rhd \blacktriangleright \bot))$$

Other seemingly unbounded phenomena arising from iteration of local processes, most importantly vowel harmony (see Charette & Göksel (1996) for a GP analysis), can be captured in a similar way. However, there are several unbounded phonological phenomena that require increased expressivity (see Graf (2009) for details). As we are only concerned with string structures, it is a natural move to try to enhance our language with operators from more powerful string logics, in particular, linear temporal logic.

The first step is the addition of two operators $\lhd^+$ and $\rhd^+$ with the corresponding relation $R^+_\lhd$, the transitive closure of $R_\lhd$. This new logic is exactly as powerful as restricted temporal logic (Cohen et al. 1993), which in turn has been shown in Etessami et al. (1997) to exactly match the expressivity of the two-variable fragment of first-order logic (see Weil (2004) for further equivalence results). Among other things, OCP effects (Leben 1973, Goldsmith 1976) can now be captured in an elegant way. The formula $O \wedge \underline{A} \wedge \overline{L} \wedge \overline{\text{ʔ}} \rightarrow \rhd^+ \neg(O \wedge \underline{A} \wedge \overline{\text{ʔ}})$, for example, disallows alveolar nasals to be followed by another alveolar stop, no matter how far the two are apart.

But $\lhd^+$ and $\rhd^+$ are too coarse for faithful renditions of unbounded spreading. For example, it is not possible to define all intervals of arbitrary size within which a certain condition has to hold (e.g. no b may appear between a and c). As a remedy, we add the until and since operators U and S familiar from linear temporal logic, granting us the power of full first-order logic. This enables us to define all star-free languages (McNaughton & Pappert 1971, Thomas 1979, Cohen 1991, Cohen et al. 1993). These feature a plethora of properties that make them very attractive for purposes of natural language processing. Moreover, the only phenomenon known to the author that exceeds their confines is stress assignment in Cairene Arabic and Creek, which basically works like the stress assignment system outlined above — with the one exception that secondary stress is not marked overtly (Mitchell 1960, Haas 1977). Under these conditions, assigning primary stress involves counting modulo 2, which is undefinable in first-order logic, whence a more powerful logic is needed. The next step up from the star-free stringsets are the regular languages, which can count modulo n. From previous research, we know that the regular stringsets are identical to the set of finite strings definable in monadic second order logic (MSO) (Büchi 1960), linear temporal logic with modal fixed point operators (Vardi 1988) or regular linear temporal logic (Leucker & Sánchez 2005). In linguistic terms, this corresponds to spreading being capable of picking its target based on more elaborate patterns.

A caveat is in order, though. Thatcher (1967) proved that every recognizable set is a projection of some local set. Thus the hierarchy outlined above collapses if we grant ourselves an arbitrary number of additional features to encode all the structural properties our logic cannot express. In the case of primary stress in Cairene Arabic and Creek, for instance, we could just use the feature for secondary stress assignment even though secondary stress seems to be absent in these languages. Generally speaking, we can reinterpret any unbounded dependency as a result of iterated local processes by using "invisible" features. Therefore, all claims about generative capacity hold only under the proviso that all such spurious coding-features are being eschewed.

We have just seen that the power of GP can be extended along the subregular hierarchy, up to the power of regular languages, and that there seems to be empirical motivation to do so. Interestingly, it has been observed that SPE yields regular languages, too (Johnson 1972, Kaplan & Kay 1994). But even the most powerful rendition of GP defines only a proper subset of the stringsets derivable in SPE, apparently due to its restrictions on the feature system, the syllable template and its government requirements. The question we face, then, is whether we can generalize GP in these regards, too, to push it to the full power of SPE and obtain a multidimensional vector space of phonological theories.

2.2. Feature Systems

The restriction to privative features is immaterial. A set of PEs is denoted by some propositional formula over $\mathscr{E}$, and the boolean closure of $\mathscr{E}$ is isomorphic to $\wp(\mathscr{E})$. But Keenan (2008, 81–109) shows that a binary feature system using a set of features $\mathscr{F}$ can be modeled by the powerset algebra $\wp(\mathscr{F})$, too. So if $|\mathscr{E}| = |\mathscr{F}|$, then $\wp(\mathscr{E}) \cong \wp(\mathscr{F})$, whence the two feature systems are isomorphic. The same result holds for systems using more than two feature values, provided their number is finitely bounded, since multivalued features can be replaced by a collection of binary valued features given sufficient co-occurrence restrictions on feature values (which can easily be formalized in propositional logic).

One might argue, though, that the core restriction of privative feature systems does not arise from the feature system itself but from the methodological principle that absent features, i.e. negative feature values, behave like constituency information and cannot spread. In general, though, this is not a substantial restriction either, as for every privative feature system $\mathscr{E}$ we can easily design a privative feature system $\mathscr{F} := \{e^+, e^- \mid e \in \mathscr{E}\}$ such that $\mathfrak{M}, w \models e^+$ iff $\mathfrak{M}, w \models e$ and $\mathfrak{M}, w \models e^-$ iff $\mathfrak{M}, w \models \neg e$. Crucially, though, this does not entail that the methodological principle described above has no impact on expressivity when the set of features is fixed across all theories, which is an interesting issue for future research.

2.3. Syllable Template

While GP's syllable template could in principle be generalized to arbitrary numbers and sizes of constituents, a look at competing theories such as SPE and Strict CV (Lowenstamm 1996, Scheer 2004) shows that the number of different constituents is already more than sufficient. This is hardly surprising, because GP's syllable template is modeled after the canonical syllable template, which in general is thought not to be in need of further refinement. Consequently, we only need to lift the restriction on the branching factor and allow theories not to use all three constituent types. SPE then operates with a single N constituent of unbounded size, whereas Strict CV uses N and O constituents of size 1. Regarding the government relations, the idea is to let every theory fix the branching factor b for each constituent and the maximum number l of licensees per head. Every node within some constituent has to be constituent licensed by the *head*, i.e. the leftmost node of said constituent. Similarly, all nodes in a coda or non-head position have to be transconstituent licensed by the head of the following constituent. For every head the number of constituent licensees and transconstituent licensees, taken together, may not exceed l.

Even from this basic sketch it should already be clear that the syllable template can have a negative impact on expressivity, but only under the right conditions. For instance, if our feature system is set up in a way such that every symbol of our alphabet is to be represented by a PE in N (as happens to be the case for SPE), restrictions on b and l are without effect. Thus one of the next stages in this project will revolve around determining under which conditions the syllable template has a monotonic effect on generative capacity.

2.4. Representations versus Derivations

One of the most striking differences between phonological theories is the distinction between representational and derivational ones, which begs the question how we can ensure comparability between these two classes. Representational theories are naturally captured by our declarative, model-theoretic approach, whereas derivational theories are usually formalized as regular relations (Kaplan & Kay 1994, Mohri & Sproat 1996), which resist being recast in logical terms due to their closure properties. For SPE, one can use a coding trick from two-level phonology (Koskenniemi 1983) and use an unpronounced feature like μ to ensure that all derivationally related strings have the same length. SPE can be then be interpreted as language over pairs and hence cast in MSO terms, which was successfully done by Vaillette (2003). Unfortunately, it is unclear how this method could be extended to subregular grammars. At the same time, no other open issue is of greater importance to the success of this project.

3. Conclusion

The purpose of this paper was to lay the foundation for a general framework in which string-based phonological theories can be matched against each other. We started with a modal logic which despite its restrictions was still perfectly capable of defining a rather advanced and intricate phonological theory. We then tried to generalize the theory along several axes, some of which readily lent themselves to conclusive results while others didn't. We saw that the power of spreading, by virtue of being an indicator of the necessary power of the description language, has an immediate and monotonic effect on generative capacity. Feature systems, on the other hand, were shown to be a negligible factor in theory comparisons; it remains an open question if the privativity assumption might affect generative capacity when the set of features is fixed. A detailled study of the effects of the syllable template also had to be deferred to later work. The most pressing issue in our opinion, though, is the translation from representational to derivational theories. Not only will it enable us to reconcile two supposedly orthogonal perspectives on phonology, but it also allows us to harvest results on finite-state OT (Frank & Satta 1998) to extend the framework to optimality theory. Even though a lot of work remains to be done and not all of our goals may turn out be achievable, we are confident that a model-theoretic approach provides an interesting new perspective on long-standing issues in phonology.

Acknowledgements

The following people had a positive influence on this paper in one way or the other: Anonymous reviewer 1 & 2, Bruce Hayes, Ed Keenan, Marcus Kracht, Ed Stabler, Kie Zuraw, and the members of the UCLA phonology seminar during the winter quarter 2009.

References

P. Blackburn, et al. (2002). *Modal Logic*. Cambridge University Press.

P. Blackburn & W. Meyer-Viol (1994). 'Linguistics, logics, and finite trees.'. *Logic Journal of the IGPL* **2**:3–29.

J. R. Büchi (1960). 'Weak second-order arithmetic and finite automata'. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik* **6**:66–92.

M. Charette & A. Göksel (1996). 'Licensing Constraints and Vowel Harmony in Turkic languages'. *SOAS Working Papers In Linguistics and Phonetics* **6**:1–25.

J. Cohen (1991). 'On the Expressive Power of Temporal Logic for Infinite Words'. *Theoretical Computer Science* **83**:301–312.

J. Cohen, et al. (1993). 'On the expressive power of temporal logic'. *Journal of Computer and System Sciences* **46**:271–294.

K. Etessami, et al. (1997). 'First-order logic with two variables and unary temporal logic'. In *Proceedings of the 12th Annual IEEE Symposium on Logic in Computer Science*, pp. 228–235.

R. Frank & G. Satta (1998). 'Optimality Theory and the Generative Complexity of Constraint Violability'. *Computational Linguistics* **24**:307–315.

J. Goldsmith (1976). *Autosegmental Phonology*. Ph.D. thesis, MIT.

T. Graf (2009). 'Comparing Incomparable Frameworks — A Model Theoretic Approach to Phonology'. To appear in the proceedings of the 33rd Penn Linguistics Colloquium. Handout version available online: http://tgraf.bol.ucla.edu/doc/talks/PLC33_mtp.pdf. Last time checked: June 14, 2009.

M. R. Haas (1977). 'Tonal Accent in Creek'. In L. M. Hyman (ed.), *Southern California Occasional Papers in Linguistics*, vol. 4, pp. 195–208. University of Southern California, Los Angeles. Reprinted in Sturtevant (1987).

J. Harris & G. Lindsey (1995). 'The Elements of Phonological Representation'. In J. Durand & F. Katamba (eds.), *Frontiers of Phonology*, pp. 34–79. Longman, Harlow, Essex.

S. Jensen (1994). 'Is ʔ an Element? Towards a Non-Segmental Phonology'. *SOAS Working Papers In Linguistics and Phonetics* **4**:71–78.

C. D. Johnson (1972). *Formal Aspects of Phonological Description*. Mouton, The Hague.

R. M. Kaplan & M. Kay (1994). 'Regular Models of Phonological Rule Systems'. *Computational Linguistics* **20**(3):331–378.

J. Kaye (1990). 'Government in Phonology: the case of Moroccan Arabic'. *The Linguistic Review* **6**:131–159.

J. Kaye (1992). 'Do you Believe in Magic? The story of s+C sequences'. *Working Papers in Linguistics and Phonetics* **2**:293–313.

J. Kaye (2000). 'A User's Guide to Government Phonology'. Unpublished manuscript. Available online: http://134.59.31.7/∼scheer/scan/Kaye00guideGP.pdf. Last time checked: June 14, 2009.

J. Kaye, et al. (1990). 'Constituent structure and government in phonology'. *Phonology Yearbook* **7**:193–231.

E. Keenan (2008). 'Mathematical Structures in Language'. Ms., University of California, Los Angeles.

K. Koskenniemi (1983). 'Two-level Morphology: A General Computational Model For Word-Form Recognition and Production'. Publication 11.

M. Kracht (1995). 'Syntactic Codes and Grammar Refinement'. *Journal of Logic, Language and Information* **4**:41–60.

M. Kracht (2003). 'Features in Phonological Theory'. In B. Löwe, W. Malzkorn, & T. Räsch (eds.), *Foundations of the Formal Sciences II, Applications of Mathematical Logic in Philosophy and Linguistics*, no. 17 in Trends in Logic, pp. 123–149. Kluwer, Dordrecht. Papers of a conference held in Bonn, November 11–13, 2000.

W. Leben (1973). *Suprasegmental Phonology*. Ph.D. thesis, MIT.

M. Leucker & C. Sánchez (2005). 'Regular Linear Temporal Logic'. In *Proceedings of The 4th International Colloquium on Theoretical Aspects of Computing (ICTAC'07)*, no. 4711 in Lecture Notes in Computer Science, pp. 291–305.

J. Lowenstamm (1996). 'CV as the Only Syllable Type'. In J. Durand & B. Laks (eds.), *Current Trends in Phonology: Models and Methods*, pp. 419–421. European Studies Research Institute, University of Salford.

R. McNaughton & S. Pappert (1971). *Counter-Free Automata*. MIT Press, Cambridge, Mass.

T. F. Mitchell (1960). 'Prominence and Syllabification in Arabic'. *Bulletin of the School of Oriental and African Studies* **23**(2):369–389.

M. Mohri & R. Sproat (1996). 'An Efficient Compiler for Weighted Rewrite Rules'. In *In 34th Annual Meeting of the Association for Computational Linguistics*, pp. 231–238.

C. Potts & G. K. Pullum (2002). 'Model Theory and the Content of OT Constraints'. *Phonology* **19**(4):361–393.

G. K. Pullum (2007). 'The Evolution of Model-Theoretic Frameworks in Linguistics'. In J. Rogers & S. Kepser (eds.), *Model-Theoretic Syntax @ 10*, pp. 1–10.

J. Rogers (1996). 'A Model-Theoretic Framework for Theories of Syntax'. In *Proceedings of the 34th Annual Meeting of the ACL*, pp. 10–16, Santa Cruz, USA.

J. Rogers (2003). 'Syntactic Structures as Multi-Dimensional Trees'. *Research on Language and Computation* **1**(1):265–305.

K. Russell (1993). *A Constraint-Based Approach to Phonology*. Ph.D. thesis, University of Southern California.

T. Scheer (2004). *A Lateral Theory of Phonology: What is CVCV and Why Should it be?* Mouton de Gruyter, Berlin.

W. C. Sturtevant (ed.) (1987). *A Creek Source Book*. Garland, New York.

J. W. Thatcher (1967). 'Characterizing Derivation Trees for Context-Free Grammars Through a Generalization of Finite Automata Theory'. *Journal of Computer and System Sciences* **1**:317–322.

W. Thomas (1979). 'Star-Free Regular Sets of ω-sequences'. *Information and Control* **42**:148–156.

N. Vaillette (2003). 'Logical Specification of Regular Relations for NLP'. *Natural Language Engineering* **9**(1):65–85.

M. Y. Vardi (1988). 'A Temporal Fixpoint Calculus'. In *Proceedings of the 15th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pp. 250–259.

P. Weil (2004). 'Algebraic Recognizability of Languages'. In J. Fiala, V. Koubek, & J. Kratochvíl (eds.), *Mathematical Foundations of Computer Science 2004*, vol. 3153 of *Lecture Notes in Computer Science*, pp. 149–175. Springer, Berlin.