# Revisions, New GM, and Zhenhai Transductions

Chris Oakden

August 12, 2018

There are four goals for this writeup:

- Offer general statements about the primitives of the Yip and Bao models, axioms about their representation
- Revise the Zhenhai I/O transduction on the Bao model
- Propose general transductions between the models
- Create a transduction between the models on the Zhenhai data

## 1 General Statements

We begin by offering some general statements about Yip's and Bao's models respectively.

Elements in Yip's model are labeled with three types of unary predicates: the syllable TBU predicate $P_\sigma$, register node $P_{rf}$ predicates, and terminal tonal node $P_{cf}$ predicates. Register and contour nodes come in two varieties:

$$\begin{aligned} P_{rf} &\overset{\text{def}}{=} P_{+u}, P_{-u} \\ P_{cf} &\overset{\text{def}}{=} P_l, P_h \end{aligned} \tag{1}$$

These correspond to upper and lower register, and High and Low tones respectively. Within a particular syllable, there is at most one syllable TBU node, one register node, and up to two terminal tonal nodes. Tones can be either level (either one 'l' or one 'h' terminal node) or contour ('h+l' for falling, 'l+h' for rising). No other terminal tonal node combinations are possible.

Within a syllable, the TBU node is the input of a binary function $\alpha(x) = y$. The output of this function is the register node.

$$\alpha(x) = y \overset{\text{def}}{=} P_\sigma(x) \wedge P_{rf}(y) \tag{2}$$

This function describes the association relation between a tautosyllabic TBU and a tonal root. Association holds between no other nodes in the model (including nodes in heterosyllabic syllables), nor is it reflexive.

The internal structure of the tone is defined by the binary function $\delta(x) = y$. This function takes the dominated element and yields its dominator. In Yip's model, terminal tonal nodes are dominated by the register node within the same syllable.

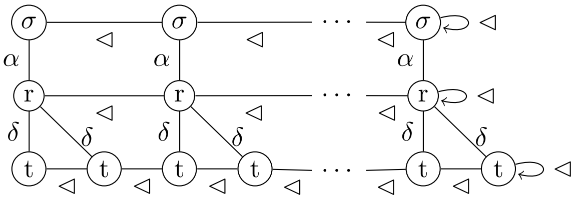
$$\delta(x) = y \overset{\text{def}}{=} P_{cf}(x) \wedge P_{rf}(y) \tag{3}$$

The dominance function is true for no other elements in the model (including in heterosyllabic syllables), nor is it reflexive.

Order is imposed on the structure via the successor function $succ(x)$. This function defines order within elements *of the same type*, that is, terminal tonal nodes within a syllable and across syllables, heterosyllabic syllable nodes, and heterosyllabic register nodes. The function is total, such that the final element in a domain is its own successor. Additionally, the successor function can be used to restrain association and dominance to single syllables, as below.

$$\begin{aligned}\alpha(x) = y &\overset{\text{def}}{=} P_\sigma \wedge P_{rf}(y) \wedge [\alpha(succ(x)) = succ(y)] \\ \delta(x) = y &\overset{\text{def}}{=} P_{rf} \wedge P_{cf}(y) \wedge [\delta(succ(x)) = succ(y) \vee \delta(succ(succ(x)) = succ(y)]\end{aligned} \tag{4}$$

The Yip model is thus:

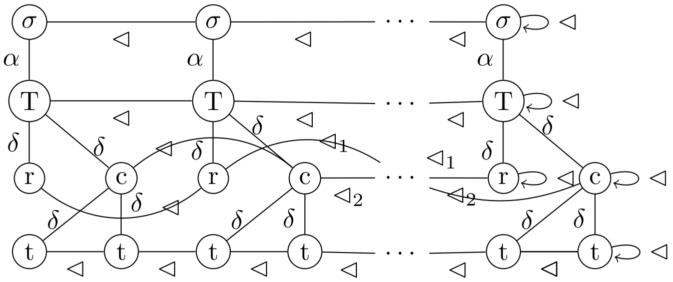


Bao's model is similar to Yip's model, with a few extensions and revisions. Two more unary predicates are necessary in addition to the original three (which are defined identically): the tonal root node 'T' predicate $P_T$, and the contour node 'c' predicate $P_c$. Within a single syllable, there is a maximum of one syllable node, one 'T' node, one register node, one 'c' node, and up to two terminal tonal nodes. Binary functions are modified such that association obtains between the syllable TBU and the 'T' node, and a different set of dominance relations. The definitions below restrict dominance and association to tautosyllabic nodes.

$$\begin{aligned}\alpha(x) = y &\overset{\text{def}}{=} P_\sigma(x) \wedge P_T(y) \wedge [\alpha(succ(x)) = succ(y)] \\ \delta(x) = y &\overset{\text{def}}{=} \big[P_{rf}(x) \wedge P_T(y) \wedge [\delta(succ(x)) = succ(y)]\big] \vee \\ &\quad \big[P_c(x) \wedge P_T(y) \wedge [\delta(succ(x)) = succ(y)]\big] \vee \\ &\quad \big[P_{cf}(x) \wedge P_c(y) \wedge [\delta(succ(x)) = succ(y) \vee \delta(succ(succ(x)) = succ(y)]\big]\end{aligned} \tag{5}$$

These functions are true only under the definitions given above, and neither function is reflexive.

Much like the Yip model, the successor function in Bao's model is defined over elements of the same type. Note that no order is imposed between register and 'c' nodes, in spite of the fact that they are sisters within the same syllable. The Bao model is thus:

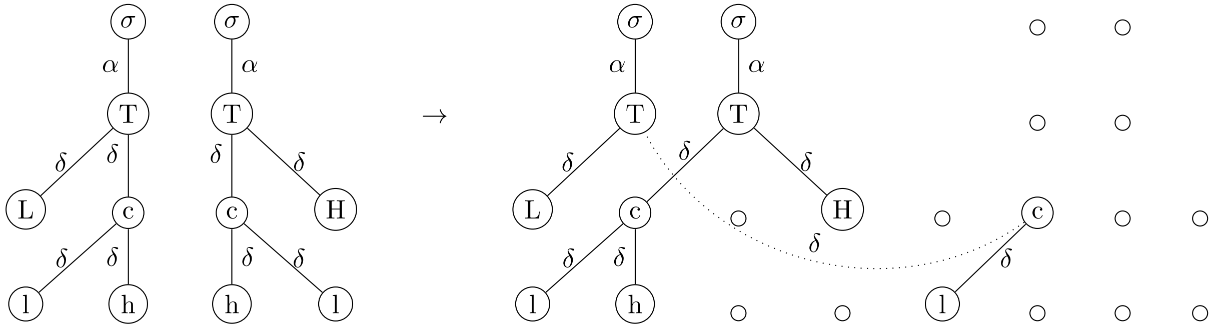


## 2 Revisions to Bao Zhenhai I/O Transduction

In this section, we revise the I/O transduction for the Zhenhai contour shift pattern using the Bao model. This transduction adopts the second alternative analysis from the previous writeup, that is, the analysis where the de-linked 'c' node from the second syllable is 'deleted' from the output. Improvements on the first pass at this transduction include new definitions for unary predicates, a new definition of the dominance function, as well as the introduction of a second copy set to account for the 'default l rule' where the 'c' node of the first syllable associates to a default terminal 'l' node. Since the 'c' node in the first copy set contains the l-h contour which associates to the second syllable, this ‘c’ node will come from the second copy set. Consider the transduction below:

$$
\begin{array}{ll}
P^1_\sigma(x) \overset{\text{def}}{=} P_\sigma(x) & P^2_\sigma(x) \overset{\text{def}}{=} \texttt{False} \\
P^1_T(x) \overset{\text{def}}{=} P_T(x) & P^2_T(x) \overset{\text{def}}{=} \texttt{False} \\
P^1_{rf}(x) \overset{\text{def}}{=} P_{rf}(x) & P^2_{rf}(x) \overset{\text{def}}{=} \texttt{False} \\
P^1_c(x) \overset{\text{def}}{=} P_c(x) \wedge \neg last(x) & P^2_c(x) \overset{\text{def}}{=} P_c(x) \wedge \neg last(x) \\
P^1_h(x) \overset{\text{def}}{=} P_h(x) \wedge \neg last(\delta(x)) & P^2_h(x) \overset{\text{def}}{=} \texttt{False} \\
P^1_l(x) \overset{\text{def}}{=} P_l(x) \wedge \neg last(\delta(x)) & P^2_l(x) \overset{\text{def}}{=} P_l(x) \wedge \neg last(\delta(x)) \\
\alpha^{1,1}(x) = y \overset{\text{def}}{=} \alpha(x) = y & \alpha^{1,2}(x) = y \overset{\text{def}}{=} \texttt{False} \\
\alpha^{2,1}(x) = y \overset{\text{def}}{=} \texttt{False} & \alpha^{2,2}(x) = y \overset{\text{def}}{=} \texttt{False} \\
\delta^{1,1}(x) = y \overset{\text{def}}{=} \Big[P_{rf}(x) \wedge P_T(y) \wedge & \delta^{1,2}(x) = y \overset{\text{def}}{=} \texttt{False} \\
\quad [\delta(succ(x)) = succ(y)]\Big] \vee & \\
\quad \Big[P_{cf}(x) \wedge P_c(y) \wedge & \\
\quad \neg last(\delta(x)) \wedge \neg last(y)\Big] \vee & \\
\quad \Big[P_c(x) \wedge P_T(y) \wedge & \\
\quad \neg last(x) \wedge last(y)\Big] & \\
\delta^{2,1}(x) = y \overset{\text{def}}{=} P_c(x) \wedge P_T(y) \wedge & \delta^{2,2}(x) = y \overset{\text{def}}{=} P_l(x) \wedge P_c(y) \wedge \\
\quad \neg last(x) \wedge \neg last(y) & \quad last(\delta(x)) \wedge \neg last(y)
\end{array}
\tag{6}
$$

Here, the ‘c’ node and its daughters from the first syllable only are produced (in the first copy set), effectively deleting the falling contour on the second syllable. A ‘c’ node with a single ‘l’ segment is produced in the second copy set to accommodate the default ‘l’ rule. Association on the output form is identical to the input (that is, only defined from the first copy set to the first copy set). The definition of $\delta^{1,1}(x) = y$ preserves the relations between register and ‘T’ nodes, while also shifting the ‘c’ node from the first syllable to the second (“contour shift”). $\delta^{2,1}(x) = y$ associates the epenthesized ‘l’ tone to the root node in the first syllable and $\delta^{2,2}(x) = y$ creates the internal structure of the epenthesized tone. The result is something like this:



One issue yet to be resolved is how the successor function is defined. The result we want is for all ordering relations to stay the same, but add an ordering where the ‘c’ and terminal nodes in the first copy set (now dominated by the second ‘T’ node) are the respective successors of the ‘c’ and ‘l’ nodes in the second copy set. I’m not sure how to do this. Is it possible that this ordering is inherited from the preserved ordering of the ‘T’ nodes themselves? I will continue to work on this.

# 3 General Transductions

In an earlier writeup, we proposed general transductions between models. Here, we revise them to accommodate polysyllabic forms. Consider the general transduction from Bao's model to Yip's model.

$$\begin{aligned}
P^{\Gamma}_{\sigma}(x) &\overset{\text{def}}{=} P_{\sigma}(x) \\
P^{\Gamma}_{rf}(x) &\overset{\text{def}}{=} P_{T}(x) \\
P^{\Gamma}_{cf}(x) &\overset{\text{def}}{=} P_{cf}(x) \\
\alpha^{\Gamma}(x) = y &\overset{\text{def}}{=} P_{\sigma}(x) \wedge P_{t}(y) \wedge [\alpha(succ(x)) = succ(y)] \\
\delta^{\Gamma}(x) = y &\overset{\text{def}}{=} P_{cf}(x) \wedge P_{T}(y) \wedge [\delta(succ(x)) = succ(y) \vee \delta(succ(succ(x))) = succ(y)] \\
succ^{\Gamma}(x) &\overset{\text{def}}{=} succ(x)
\end{aligned} \tag{7}$$

This transduction captures the equivalency between Bao's T node and Yip's register node as the tonal root. Additionally, the definitions of association and dominance account for monosyllabic and polysyllabic structures.

Translating from the Yip model to the Bao model requires a transduction with a copy set of size two. Recall that in an earlier writeup, this strategy was also used to derive the structural similarities between the register node in Yip's model and both the 'T' and 'c' nodes in Bao's model. Working from that initial attempt, we can append the association and dominance functions to limit these relations to tautosyllabic nodes as we have done above. However, this will not work.

$$\begin{aligned}
P^{1}_{\sigma}(x) &\overset{\text{def}}{=} P_{\sigma}(x) & P^{2}_{\sigma}(x) &\overset{\text{def}}{=} \texttt{False} \\
P^{1}_{T}(x) &\overset{\text{def}}{=} P_{rf}(x) & P^{2}_{T}(x) &\overset{\text{def}}{=} \texttt{False} \\
P^{1}_{rf}(x) &\overset{\text{def}}{=} \texttt{False} & P^{2}_{rf}(x) &\overset{\text{def}}{=} P_{\sigma}(x) \\
P^{1}_{c}(x) &\overset{\text{def}}{=} \texttt{False} & P^{2}_{c}(x) &\overset{\text{def}}{=} P_{rf}(x) \\
P^{1}_{cf}(x) &\overset{\text{def}}{=} \texttt{False} & P^{2}_{cf}(x) &\overset{\text{def}}{=} P_{cf}(x) \\
\alpha^{1,1}(x) = y &\overset{\text{def}}{=} P_{\sigma}(x) \wedge P_{rf}(y) \wedge & \alpha^{1,2}(x) = y &\overset{\text{def}}{=} \texttt{False} \\
& \quad [\alpha(succ(x)) = succ(y)] & & \\
\alpha^{2,1}(x) = y &\overset{\text{def}}{=} \texttt{False} & \alpha^{2,2}(x) = y &\overset{\text{def}}{=} \texttt{False} \\
\delta^{1,1}(x) = y &\overset{\text{def}}{=} \texttt{False} & \delta^{1,2}(x) = y &\overset{\text{def}}{=} \texttt{False} \\
\delta^{2,1}(x) = y &\overset{\text{def}}{=} \big[P_{\sigma}(x) \wedge P_{rf}(y) \wedge & \delta^{2,2}(x) = y &\overset{\text{def}}{=} P_{cf}(x) \wedge P_{rf}(y) \wedge \\
& \quad [\delta(succ(x)) = succ(y)]\big] \vee & & \quad [\delta(succ(x)) = succ(y) \vee \\
& \quad \big[P_{rf}(x) \wedge P_{rf}(y) \wedge & & \quad \delta(succ(succ(x))) = succ(y)] \\
& \quad [\delta(succ(x)) = succ(y)]\big] & &
\end{aligned} \tag{8}$$

The problem here is in the part of the definitions of the dominance function that ensure that dominance only occurs within syllables and not between syllables. In defining unary predicates on the second copy set, we use structural positions that are not identical (or even equivalent) to the input. Specifically, we define the register node in the structural position of the syllable. We refer to the definition of $\delta(x) = y$ in the input structure along with $succ(x)$, but given the structural mismatches, sometimes those relations *are not available* since they are not defined that way in the input structure. This is not an issue for the definition of $\delta^{2,2}(x) = y$, but it is for $\delta^{2,1}(x) = y$. There is no such definition of dominance between syllable and register in the input model (perhaps we can change it to $\alpha(succ(x)) = succ(y)$ since that relation does hold in the input structure), and certainly no definition of dominance between register nodes. I have yet to reach a satisfying solution to this problem, and will continue to work it out.

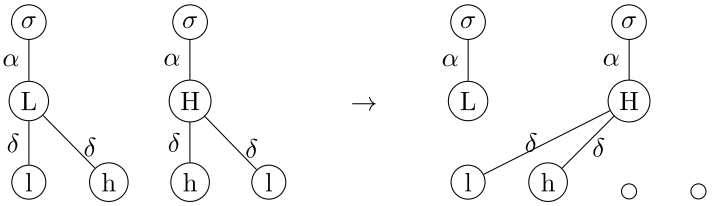
# 4 Transductions between models: Zhenhai

We have shown that we can perform transductions to translate from Bao's model to Yip's model and vice versa, and that the complexity of these transductions is QF. We have also shown that we can represent contour spreading in Zhenhai using Bao's model, in accordance with previous analyses. The data from Zhenhai are used as an example of a pattern which is impossible to represent in Yip's model. However, if there is some equivalency between the two (such that one can translate between them using simple logics), should there be processes that can be formalized in one model but not the other?

I have considered a number of approaches to doing a 'transduction between the two models' on the Zhenhai data, but haven't stopped to think about what that actually means, and what the possibilities are. One is a transduction on a Zhenhai input from Bao's model to Yip's model. Another is the same, but on outputs. Neither would capture what is of interest to us, which is the *process* itself. So, I will first attempt to model contour spreading with an I/O contour using Yip's model. This is presented in the transduction below:[1]

$$\begin{aligned} P^{\tau}_{\sigma}(x) &\overset{\text{def}}{=} P_{\sigma}(x) \\ P^{\tau}_{rf}(x) &\overset{\text{def}}{=} P_{rf}(x) \\ P^{\tau}_{l}(x) &\overset{\text{def}}{=} P_{l}(x) \wedge \neg last(\delta(x)) \\ P^{\tau}_{h}(x) &\overset{\text{def}}{=} P_{h}(x) \wedge \neg last(\delta(x)) \\ \alpha^{\tau}(x) = y &\overset{\text{def}}{=} \alpha(x) = y \\ \delta^{\tau}(x) = y &\overset{\text{def}}{=} P_{cf}(x) \wedge P_{rf}(y) \wedge \neg last(\delta(x)) \wedge last(y) \end{aligned} \tag{9}$$

The definitions are straightforward. The intuition here is to preserve only the 'l' and 'h' tones (the rising contour) on the first syllable while deleting the tones from the second syllable. This is achieved by adding a second conjunct $\neg last(\delta(x))$ to the unary relations on terminal tonal nodes. Then, the dominance function is defined such that these tones are dominated by the register node on the final syllable. Note that this definition is similar to that proposed for the same transduction using Bao's model (both the version in this writeup and the previous version).



We have shown that contour spread *can* be represented using Yip's tonal geometry. Additionally, we have defined the dominance function in the transduction such that the contour on the first syllable spreads in a *unit-like* manner (by referring to the structural position of the 'dominator' in the input, but not to its substance, i.e., low register). In other words, it is possible to spread contour without spreading register in Yip's model, in spite of the fact that register immediately dominates terminal nodes.

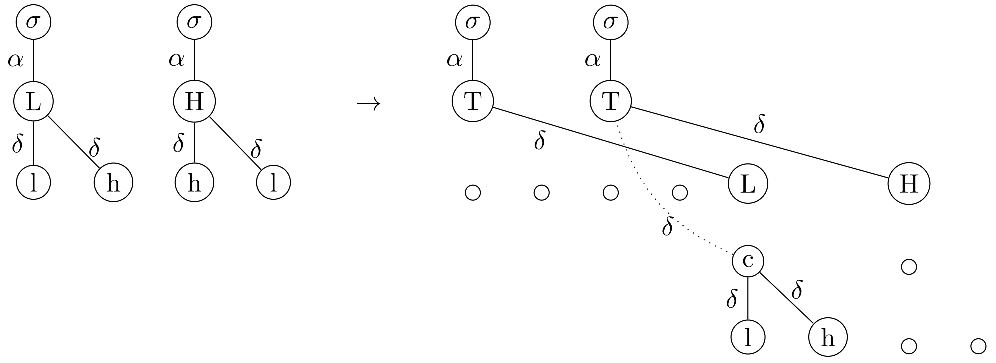
Another option is to do a Zhenhai transduction from a Yip model input to a Bao model output. As was the case in the general transduction, this will require a copy set of size two. We follow the convention of the general model by defining the TBU and tonal root nodes in the first copy set only (thereby allowing an identical definition of association), with register, 'c', and terminal tonal nodes defined in the second copy

[1] I am abstracting away from the default rule for now. Suffice it to say that the default insertion of the 'l' tone would proceed in the same way as for the Bao model in section 2: adding a second copy of a terminal tonal 'l' node from the first syllable, and then defining domination from the second copy set to the first such that 'l' in copy 2 is dominated by the low register node from copy 1.

set. This transduction is presented below:

$$\begin{array}{rlrl}
P^1_\sigma(x) \overset{\text{def}}{=} & P_\sigma(x) & P^2_\sigma(x) \overset{\text{def}}{=} & \texttt{False} \\
P^1_T(x) \overset{\text{def}}{=} & P_{rf}(x) & P^2_T(x) \overset{\text{def}}{=} & \texttt{False} \\
P^1_c(x) \overset{\text{def}}{=} & \texttt{False} & P^2_c(x) \overset{\text{def}}{=} & P_{rf}(x) \wedge \neg last(x) \\
P^1_{-u}(x) \overset{\text{def}}{=} & \texttt{False} & P^2_{-u}(x) \overset{\text{def}}{=} & P_\sigma(x) \wedge \neg last(x) \\
P^1_{+u}(x) \overset{\text{def}}{=} & \texttt{False} & P^2_{+u}(x) \overset{\text{def}}{=} & P_\sigma(x) \wedge last(x) \\
P^1_l(x) \overset{\text{def}}{=} & \texttt{False} & P^2_l(x) \overset{\text{def}}{=} & P_l(x) \wedge \neg last(\delta(x)) \\
P^1_h(x) \overset{\text{def}}{=} & \texttt{False} & P^2_h(x) \overset{\text{def}}{=} & P_h(x) \wedge \neg last(\delta(x)) \\
\alpha^{1,1}(x) = y \overset{\text{def}}{=} & \alpha(x) = y & \alpha^{1,2}(x) = y \overset{\text{def}}{=} & \texttt{False} \\
\alpha^{2,1}(x) = y \overset{\text{def}}{=} & \texttt{False} & \alpha^{2,2}(x) = y \overset{\text{def}}{=} & \texttt{False} \\
\delta^{1,1}(x) = y \overset{\text{def}}{=} & \texttt{False} & \delta^{1,2}(x) = y \overset{\text{def}}{=} & \texttt{False} \\
\delta^{2,1}(x) = y \overset{\text{def}}{=} & \Big[P_\sigma(x) \wedge P_{rf}(y) \wedge & \delta^{2,2}(x) = y \overset{\text{def}}{=} & \Big[P_l(x) \wedge P_{rf}(y) \wedge \\
& \neg last(x) \wedge \neg last(y)\Big] \vee & & \neg last(\delta(x)) \wedge \neg last(y)\Big] \vee \\
& \Big[P_\sigma(x) \wedge P_{rf}(y) \wedge & & \Big[P_h(x) \wedge P_{rf}(y) \wedge \\
& last(x) \wedge last(y)\Big] \vee & & \neg last(\delta(x)) \wedge \neg last(y)\Big] \\
& \Big[P_{rf}(x) \wedge P_{rf}(y) \wedge & & \\
& \neg last(x) \wedge last(y)\Big] & &
\end{array} \tag{10}$$

Graphically, this transduction is:


σ
α
L
δ
δ
l
h
σ
α
H
δ
δ
h
l
→
σ
α
T
σ
α
T
δ
δ
L
H
δ
c
δ
δ
l
h