CLINICAL LINGUISTICS & PHONETICS, 1988, VOL. 2, NO. 2, 115–137

TUTORIAL REVIEW

Automating the analysis of child phonology†

CLIFTON PYE

Department of Linguistics, University of Kansas, Lawrence, Kansas 66045, USA

and

DAVID INGRAM

Department of Linguistics, University of British Columbia, Vancouver, British Columbia V6T 1W5, Canada

(*Received 14 January 1988; accepted 1 February 1988*)

Abstract

Ingram (1981, in press) introduced a set of procedures for phonological analysis of fundamental importance to research on phonological development and the clinical assessment of phonological disorders. Recent studies employing these procedures demonstrate their utility in research on the course of normal development in English (Ingram, 1981), phonological disorders (Ingram, 1986; Pye, Wilcox and Siren, in press) and cross-linguistic comparisons of phonological development (Ingram, 1981/2, 1983; Pye, Ingram and List, 1987).

Despite such theoretical and practical successes, the procedures remain relatively unutilized in clinical practice. This is primarily due to the exhaustive analysis of language samples that the procedures demand. Most of the procedures can be automated, however, which should provide a greater incentive for their use in clinical settings. This report provides an overview of the analytic procedures and demonstrates how the procedures are performed in the PAL implementation. It also addresses the programming decisions which are an integral part of the implementation and which give the programme its flexibility and power.

Introduction

In the last decade, the phonological analysis of delayed children has finally risen above the use of simplistic articulation tests. This change can be seen in either the abandonment altogether of articulation tests in favour of spontaneous language samples, or the analysis of the test results by a more sophisticated linguistic instrument. The number of suggested analytic procedures has also increased, with

† This paper was originally presented to the American Speech, Language and Hearing Association meeting, held on 14 November 1987.

occurred. PAL, however, does allow the user to make their own decisions in this regard.

Another decision in the phonetic analysis regards when we can say that a child has or 'uses' a particular sound. Ingram (1981) suggests setting some *Criterion of Frequency* for making such a decision. Sounds will then fall into four natural categories in regard to this criterion—(1) Never used; (2) Marginal (or Transitional), i.e. they occur but below the criterion; (3) Used, i.e. sounds that meet or are moderately above the criterion; and (4) Frequent, i.e. sounds that occur substantially above the criterion. Individual analysts can of course adjust these according to their own preferences. Individual choice also arises regarding how the Criterion of Frequency is set. Ingram (1981, in press) provides one suggestion in this regard. For English, it is proposed that consonants should occur at least once for each 25 words which are collected. This is reached on the rough calculation that there are approximately 25 English consonants, and that each should occur at least once in 25 words if they are distributed randomly. Such a measure, of course, becomes more secure as sample sizes are increased. We find samples of at least 75 words are reasonable for using this measure, although it can be adapted for smaller samples.

The second step is the determination of the child's *matches* and *substitutions*. This is done for both consonants and vowels, and involves comparing the child's segments with their targets in the adult words. Ingram (in press) discusses two basic distinctions in this regard, i.e. *lexical* versus *nonlexical* matches and substitutions. A lexical match or substitution is one which only occurs for one word. For example, suppose the child's only attempt at the initial consonant /k/ is in 'cat', and it is pronounced [ka]. The child in this case has a lexical match for /k−/. Nonlexical matches and substitutions are those instances where there are data for more than one word. Ingram (in press) further proposes a measure for when we can say that a child has acquired a sound. It is when a child shows matches for a particular adult sound in *over 50%* of its attempts of the sound. This is an extremely simple measure, but in practice it has proved to be insightful (if used with discretion).

The matches which result from the substitution analysis provide an insight into those sounds which the child has acquired. The substitutions, on the other hand, show where errors occur. As has been noted for several years, these errors often pattern together rather than appear idiosyncratically. When such patterns occur, they can be described by what have been referred to as *phonological processes*. A number of suggestions exist on what constitutes the set of possible phonological processes, and we will not go into this controversial point here. PAL, however, leaves the phonological process analysis up to the individual analyst.

The last step in the phonological analysis is the determination of the child's contrasts. This is the result of comparing the results of the phonetic analysis with the substitution analysis, and applying one's phonological theory to the data. At the simplest level, we can say that a child has a particular phoneme when it is phonetically used and occurs as a nonlexical match. At a more advanced level, we can introduce distinctive features and propose that certain phonological oppositions exist if they share certain features. One such proposal appears in Goad and Ingram (1987) where we suggest that 'two phonemes form an opposition when they have opposite values for no more than one (nonredundant) feature'. For example, if a child has /b/ and /t/ only among the stops, he does not have the feature [voice], while the child with /b/ and /p/ does. We also introduce the notion of also considering the perceptual system of the child, a point first suggested in Ingram (1983). On such points, PAL leaves the actual decision up to the analyst.

PAL: the microcomputer implementation

Given these introductory remarks about the form of the phonological procedures, we can now examine PAL. The discussion will consist of two parts: (1) a demonstration of a microcomputer implementation of the above procedures; and (2) a discussion of the programming decisions that accompanied the implementation. The programming decisions are an integral part of the programs since a programmer cannot foresee every eventuality that might arise in the course of using the programs. This inevitably leads to some unforeseen limitations on the part of the program. It is just as well to be as fully aware as possible of the program's advantages and limitations in order to decide whether to use the program or do the analysis by hand. We hope you will gain a clear understanding of the advantages and limitations of the PAL implementation as well as a working knowledge of the program.

Those of you who have some familiarity with computer programs know that although computers can sort pages of data in the flash of an eye, they are also extremely stupid. They only do exactly what they are told to do, which is often not what you wanted. The first step in using any program is to make sure the computer knows where to find the information it needs in order to do the analysis. This means that the children's language samples must be entered into the computer in the right style or format. Otherwise, the computer wouldn't be able to tell the difference between the child's speech and the mother's.

Most of you are probably acquainted with one or another of the word-processing programs on the market. Wordstar is probably the most widely known, but there are others such as Microsoft Word, and Word Perfect. What you may not know is that many word-processing programs insert symbols into the document you are typing so that they can keep track of paragraphs, line feeds, printer commands, etc. These symbols don't appear on the computer screen, but they are part of the text on the computer disk. These symbols won't affect the text as long as you keep using the same word-processing software. They will, however, cause problems for programs like PAL which are only designed to read the usual set of alphabetic characters.

Thus, the first step in using PAL is to find a word-processor that does *not* insert special characters into the text. In computerese this means the text should be produced in what is known as the ASCII format. Most word-processors these days have a way of producing an ASCII text. Wordstar, for example, can be used in the non-document mode. We recommend trying PC-WRITE, a Shareware word-processor that is simple to operate and produces a pure ASCII text. PC-WRITE even has a way of converting Wordstar files to ASCII files by typing 'Shift 5'. You can make use of the computer's operating system to perform a quick test to see if your word processor has produced an ASCII text or not. In the initial system mode, where you see the prompt 'A〉', type the command 'type sample.doc' where *sample.doc* is the name of the transcript you have entered. This will scroll through the text on the monitor. If the text appears exactly as you typed it in, then it is in the ASCII format. However, if you see strange characters and hear bells, then your word-processor does not produce a pure ASCII text.

Transcription

Once you have settled on a way of getting the information into the computer you need to make sure that the language sample is typed in the right style or format. Transcripts to be analysed with the PAL routines should follow certain conventions (otherwise the computer will get lost). In general, the PAL conventions follow those used in the SALT programs (Miller and Chapman, 1985). For one thing, we need to have a way of telling the computer how to find the child's speech. The PAL/SALT convention is to begin the transcript with a name line that is marked by the dollar sign '$' in the first column. The child's name should appear first, followed by any other participants, as in (2). (An expanded version of this sample can be found in Appendix A.)

(2) *Sample of data entered in PAL*

```
$  Daniel, Father
+  Age: 2;0(18).
+  Context: Bookreading.
D  hI:bVbgagi. (phonetic tier)
   〈      〉.  (orthographic tier)
F  What, honey?
F  What's this here?
D  767 kHIpH. (each tier has a punctuation mark)
   a tape.
F  Taperecorder.
D  kHipHukHOr6.
   taperecorder. (words combined to show unproductive forms)
```

The program automatically assumes that the first name is the one that you are interested in. Furthermore, it assumes that every line of the child's speech sample will begin with the first letter of the first name on the name line. In this case, every line of Daniel's speech should be identified with the capital letter 'D'.

At this point we should say a word about the general nature of the child's language sample. It is only necessary to put as much information as you need into the transcript for this purpose. In practice this will mean that you only need to enter the speech from the subject in whom you are interested. PAL, however, contains procedures for morphological and syntactic analyses as well as the phonological analysis. It is also flexible enough to handle transcripts with several speakers. This means that you can always add more information to the transcript as time and interest warrant.

PAL permits the child's utterances to have any number of lines (or tiers), as long as all of the child's utterances in the transcript have the same number of tiers. The entire utterance should be typed on a single line. The sample transcript in Appendix A contains two tiers per utterance. The first tier should contain the *phonetic transcription* while the second tier should contain the *orthographic transcription*. If desired, further tiers may be added to code morphological, syntactic, semantic or whatever information is appropriate. Successive tiers for the child should not begin with the child's initial. Only the child's utterances may have more than one tier associated with each utterance.

Every line in the transcript should end with a punctuation mark (.!? or ...). We use the succession of periods to indicate discontinued or interrupted utterances. The orthographic tier follows most of the conventions suggested in Ingram (in press). Unintelligible parts of utterances have empty angle brackets ('〈 〉') on the orthographic tier. Inaudible or indistinguishable parts of an utterance should also be marked with '〈 〉' on the phonetic tier. Questionable parts of utterances have the questionable part of the interpretation marked with angle brackets (for example

⟨truck⟩) in the orthographic or phonetic tier. The transcript does not distinguish between stuttered words and repeated words. Both are marked with an ampersand ('&') at the beginning of each repeated attempt. Repeated words are not included in any of the analyses.

The transcript should follow Ingram's (in press) Principle of Phonetic Correspondence in that every word on the orthographic tier should have a partner on the phonetic tiers. The words should be separated in the same way on each tier (i.e. by a slash or space).

One problem you soon run into in entering phonetic transcriptions into the computer is that the standard set of ASCII characters does not include all of the IPA symbols. In fact the characters that are available on computers are fairly limited. There are word processors available that allow you to design your own set of characters, but these are not widely available. Also, there are not any established conventions on which IPA symbols should be assigned to which keys.

PAL gets around this problem by using a translation of the IPA alphabet that uses the standard set of ASCII characters. This is shown in (3). Most of the translations are fairly obvious, but a few are not immediately apparent. The /6/ is a schwa, the /F/—a flap, and the ampersand /&/—the rhoticized schwa. By combining upper and lower case letters and numbers you can do a reasonable job of representing the necessary characters for the phonetic transcription of English. PAL assumes that the phonetic tier has been typed using this alphabet. You can use any other characters you like in the transcription, but they won't be included in the phonological analysis. PAL avoids using two characters to write one sound as much as possible since such digraphs add to the amount of work the computer program must accomplish.

(3) *An ASCII IPA*

Consonants

p		t	c	k	7
b		d	j	g	
m		n		N	
	f	T	s	S	h
B	v	D	z	Z	
			l		
	f	F, &			
W	w		y		

Vowels

i		u
I	6	U
e		o
E	V	O
A	a	
ay	6y	oy
aw		

c —voiceless, palatal affricate
j —voiced, palatal affricate
T —voiceless, interdental fricative
D —voiced, interdental fricative
S —voiceless, palatal fricative
Z —voiced, palatal fricative
B —voiced, bilabial fricative
F —flap
W—voiceless, bilabial glide

Special symbols

H—aspiration
7 —glottal stop
&—rhoticised schwa
6 —schwa
y —glide
w—glide
: —length
~—nasalization

Phonological analysis

Once the transcript is phonetically and orthographically transcribed onto the computer in an ASCII format the phonological analysis is relatively straightforward. This is because the computer can now do what it does best—sort through every syllable of the child's pronunciation.

The first step in the analysis is to set up the *phonological lexicon*, i.e. an alphabetical list of the words attempted by the child along with their pronunciation. The PAL routine keys its analysis to the words in the orthographic tier. This is the reason it is important to observe the Phonetic Correspondence Principle throughout the transcription. The resulting analysis lists each word on the orthographic tier, the number of times the word occurs in the transcript, the utterances in which the word appears, and the way the child pronounced each token. An excerpt of the output of the lexical productivity routine is shown in (4). (A complete analysis of this kind for the sample in Appendix A is given in Appendix B.)

(4) *A sample output of the first step in PAL*
Daniel's utterances:

1. hI:bVbgagi.
   ⟨ ⟩.
2. 767 kHIpH.
   a tape.
3. kHipHukHOr6.
   taperecorder.

*Number of Daniel's lines = 50
*Total number of lines = 106

Daniel's word index:

WORD	FREQ	SENTENCES
a:	5	2, 12, 14, ⟨22⟩, 23,
airplane:	1	18,
airplane/s:	1	21,
and:	2	22, 23,
another:	1	32,
banana:	2	14, 15,
basket:	1	50,

Daniel's phonological lexicon

a	767 6 n: ⟨6⟩ 6
airplane	eytHeyn
and	eynd 6d
another	n6nE

basket	bAsAtH
boat	bowtH
	bowt
	bowt
book	⟨sebawu:kH⟩
	bEkH
	b6k
	bEkH

The program preserves the angle brackets around each tier so that you can tell how often each word appears in a questionable interpretation or partly inaudible transcription. For example, one of the phonetic tokens of *a* in Daniel's phonological lexicon is enclosed in brackets to indicate that it has a questionable interpretation. Looking up the word in Daniel's word index shows that the questionable interpretation occurred in utterance 22, and that it was the only one of five phonetic tokens that was questionable.

In practice there are a number of things that can go wrong at this point in the computer analysis. These are usually due to transcription errors of one sort or another. If you have forgotten to put in a punctuation mark at the end of one of the lines, the computer will assume it continues onto the next line. If you don't have the same number of words in the phonetic and orthographic tiers, the computer will assign the wrong phonetic form to each word from then on. One particularly annoying bug is the lack of a linefeed at the end of a line. This halts the computer output in the middle of the list, just before the problematic word.

PAL includes a checking routine which looks for such obvious errors as the number of tiers associated with each of the child's utterances and punctuation at the end of each line. However, it cannot check for correct phonetic transcription or the right correspondence between the phonetic and orthographic tiers without adding a dictionary in which it could look up the pronunciation of each word. Since children don't pronounce every word in the same way this is impossible.

A glance through the phonological lexicon should tell you if there are any problems with the correspondence between the phonetic and orthographic tiers. This would result in a phonetic token like [hI:bVbgagi] for the word *airplane*. If the phonological lexicon doesn't include any words that begin with a letter after *s*, there is a line in the transcript somewhere that doesn't have a linefeed after it. This can be corrected by going back to the transcript and hitting the RETURN key at the end of the line. With a little practice you should be able to quickly track down the source of the problem.

The abbreviated phonological lexicon

Assuming the phonological lexicon procedure has worked, the next step is to produce an *abbreviated phonological lexicon*. This is done by eliminating questionable phonetic types from the analysis and selecting the most representative phonetic type for each lexical type.

This is actually the most difficult part of the analysis to computerize. This is because the most representative phonetic type is not always immediately obvious. Ingram (1981) gives four steps for selecting this type. The first, selecting the phonetic type that occurs in a majority of the phonetic tokens, is simple and easy to implement. The second, selecting the phonetic type that shares the most segments with the others, is *much* harder to implement. To even begin such a comparison, the computer would have to be instructed on how to align the words so that it could count the number of corresponding segments. An example shows what this can involve. Ferguson and Farwell's (1973) data from T at VI shows 4 spontaneous phonetic types for the lexical type *shoe*

(5) *Lexical type*	*Phonetic type*
shoe	1. ISu
	2. SIu
	3. Su
	4. tHuH

If the computer simply proceeded from left to right on a segment by segment comparison it would match the initial [I] of the first phonetic type with the initial [S] of the second. But it seems clear that the [S] of the first type should be matched to the [S] of the second. This could be done by having the computer first of all look for identical sounds in the phonetic types. But this still wouldn't enable the computer to match the initial [tH] of the fourth phonetic type with the [S] of the first.

The only solution would be to supply the computer with a dictionary of phonetic features that would allow it to first search for corresponding consonants and then check for consonants which share several features, say voice and place. Such a procedure would have to be very sophisticated indeed to handle complex cases of metathesis and deletion which are common in children's attempts at long words such as 'elephant' or 'struthiomimus'. In the end Pye was not convinced that it would be possible to implement a procedure that could be guaranteed to work correctly in 100% of the children's data. Even more limiting would be the fact that all this programming would be devoted to comparing phonetic forms in English, and Pye was interested in developing a program that could be used with any language.

PAL therefore has the user produce their own abbreviated phonological lexicon by hand. Actually, this turns out to be quite easy, given that the computer has already produced a complete phonological lexicon for the child. To begin, you copy the existing phonological lexicon into a new file using the word processor. Then you can edit this new file in the same way you would edit an ordinary document. Following the procedures outlined in Ingram (1981), you next select the most representative phonetic token for each lexical type. Then delete all of the other phonetic tokens for that lexical type. Once you become familiar with this part of the procedure the actual editing can be done very rapidly. A further advantage of doing this part of the procedure by hand is that you become more familiar with the data and can notice things that might escape the phonological analysis.

The result of this step will be an abbreviated phonological lexicon that shows the child's most representative phonetic type for each lexical type. Before the computer can begin the phonological analysis, however, you need to tell it how an adult would produce each lexical type. PAL contains a procedure which looks up the presumed target in a file *phonetic.dic* and writes the adult phonetic form onto the file containing the abbreviated phonological lexicon. This routine also indicates the words it was unable to find. The resulting file can then be edited with a word processor to make certain that each line contains a word in standard orthography, the phonetic target, and the child's phonetic type. A sample abbreviated phonological lexicon is shown in (6).

(6) *Daniel's abbreviated phonological lexicon*

Orthographic form	*Phonetic target*	*Child's form*
a	6	6
airplane	eyrpleyn	eytHeyn
and	And	eynd
another	an6D&	n6nE
banana	b6nAn6	nAnA
basket	bAskHEt	bAsAtH
bike	baykH	BeykH
boat	bowt	bowt
book	bUk	bEkH
call	kHal	kHow
car	kHar	kHuA
corn	kHorn	kown6
dance/ing	dAnsiN	dAnsin
it	It	In
lace/s	lays6z	maysEn
let/'s	lEts	E7

(A complete one for the sample in Appendix A is found in Appendix C.)

The phonetic inventory

Once you have completed the abbreviated phonological lexicon you are ready for the phonological analysis. The first step is to sort all of the child's phonetic types by their initial and final consonants and vowels. PAL includes a routine that does this for you. A sample output is shown in (7). Note that the routine breaks consonant clusters into separate segments and displays the child's phonetic type separately under each segment in the cluster. In the final consonants, for example, the phonetic type [kHats] would occur under both the [t] and the [s].

(7) *Sample phonetic inventory*

Phonetic inventory

p	t	c	k
pHiFu peytH	tHE:n		kHow kHuA kown6 kHey kHVd6
i	I	e	E
pHiFu si: kHipHukHOr6	In kHIpH		bEkH E7 tHE:n
p	t	c	k
kHIpH	bAsAtH bowt peytH kHats		BeykH bEkH

This output permits a rapid determination of the frequency of the child's sounds in each phonological context. The frequency of each sound is compared with a Criterion of Frequency (CF), as discussed earlier. To produce a phonetic inventory, you only have to print out this output with your word processor and circle the used sounds, enclose the marginal sounds in parentheses, star the frequent sounds, and cross off the unused sounds. (Appendix D contains a more complete analysis.)

The substitution analysis

The next step in Ingram's procedure is an analysis of the child's patterns of substitution. The PAL substitution analysis displays a phonetic inventory of the adult targets with the child's production beside each one. These are exemplified in (8). (See Appendix E for a complete analysis.)

(8) *Sample of the substitution analysis*

Substitutions

p	t	c	k
pHiF& pHiFu	tHeyp kHIpH		kHal kHow
pHig peytH	tHeyprikHord&		kHar kHuA
	kHipHukHOr6		
plein tHE:n	trVks kHats		kHorn kown6
i	I	e	E
pHiF& pHiFu	It In	plein tHE:n	1Ets E7
pHig peytH	1IF61 Iy6		
rikHord& kHVd6			
si: si:			
p	t	c	k
tHeyp kHIpH	bAskHEt bAsAtH		baykH BeykH
	bowt bowt		bUk bEkH
	It In		trVks kHats
	1Ets E7		
	nat now		
	want n6		
	wVts h676		

This output again makes it easy to determine which of the child's sounds match the adult target and which sounds the child uses as substitutions for specific targets. You can print out this analysis and simply circle the child's sounds that match the adult target and enclose the child's marginal matches in parentheses.

It is worth pointing out again that although this seems like a trivial process to do by hand, it would actually take a highly sophisticated computer routine to accomplish the same task. We run into exactly the same problem we encountered before in figuring out how to program the computer to line up the child's forms with the adult target in a way that the computer could figure out whether the child actually produced a sound that matches the initial or final consonant of the adult form. We might also emphasize that doing this part by hand is done rather quickly, a stark contrast to the amount of time needed to do the kinds of sorting done by PAL.

The phonological inventory

The substitution analysis is the next to the last step in determining the child's phonological inventory. To produce this inventory, you combine the information from the child's phonetic inventory and substitution patterns as discussed earlier. Since all of the necessary information is already available in the phonetic inventory and substitution analysis, this is relatively easy to do. More sophisticated analyses, of course, will depend on the training of the analyst.

Using PAL

The PAL routines were written in Spitbol, a compiled version of the Snobol4 programming language (cf. Griswold, Poage and Polonsky [1968] *The Snobol4 Programming Language*). Spitbol runs on IBM compatible microcomputers using the DOS operating system with at least 256K of RAM. There are actually three programs: *formix.exe*—which checks the transcript's format and converts SALT and ChilDes formats to the PAL format, *phonix.exe*—which does the phonological analysis, and *syntax.exe*—which performs the morphological and syntactic analyses.

To run the phonological analysis program you need only insert the disk with the child's language sample in one disk drive and the PAL disk in the other drive. Change to the drive with the PAL disk and type: phonix [RETURN]. The phonological routine will load into the computer's memory and display a menu listing the available options. This menu is shown in (9):

(9) *Menu used by Phonix*

```
SSSSSSSS  SS        SS   SSSSSSSS   SS        SS  SSSSSSSS  SS        SS
SS     SS SS        SS  SS        SS S SS     SS     SS       SS     SS
SS     SS SS        SS  SS        SS SS SS    SS     SS        SS   SS
SSSSSSSS  SSSSSSSSSSSS  SS        SS SS  SS   SS     SS          SSSS
SS        SS        SS  SS        SS SS   SS  SS     SS        SS   SS
SS        SS        SS  SS        SS SS    SS S      SS       SS     SS
SS        SS        SS   SSSSSSSS   SS        SS  SSSSSSSS  SS        SS
```

(c) 1986 by C. Pye
200 Arrowhead
Lawrence, KS 66044

Please select option:

1. Create phonological lexicon.
2. Phonological analysis.
3. Quit.

Type in a number (1–3) and return.

If you were to type number '1' the computer would then ask for the name of the file with the child's language sample which might be typed in as:

b:daniel.dat

Then you will be prompted for the name you wish to give the file for the phonological lexicon, which might be:

b:daniel.lex

Finally, the computer will ask how many tiers are associated with each of the child's utterances. In our example it is 2. The program will then write the child's phonological lexicon on the disk in the file named daniel.lex. This makes it possible to directly incorporate the analysis into texts or records you might produce about a subject. This file may be altered any way you please with a word processor. In particular, you may edit the phonological lexicon to produce an abbreviated phonological lexicon for the phonological analysis procedure. Once you have accomplished this, you can then return to the phonix program by typing phonix and choose option 2. You can then print out the resulting inventories with your word processor.

In the course of using PAL, you may find that it is easier to keep track of the different analyses by following a systematic procedure when naming the output files. PAL suggests two alternatives. One uses the subject's name as the file name and a code in the extension for the contents of the file. The alternative is to name the content of each file, and use the extension for the subject identification. Both of these are exemplified in (10):

(10) *Some conventions for naming files in PAL*

File contents	*File name*	or	*File name*
language sample	name.dat		sample.d
checking routine	name.chk		check.d
lexical productivity	name.lex		plexicon.d
abbreviated lexicon	name.abv		alexicon.d
phonological analysis	name.phn		phone.d
mlu and summary statistics	name.mlu		mlu.d
sentential productivity	name.sen		sentence.d
syntactic productivity	name.syn		syntax.d

We are still testing the PAL procedures and cannot say what the maximum length of the language sample might be that PAL can analyze. Pye has used it successfully with files with a length of 20K, which corresponds to roughly 20 pages of a typed transcript. This would normally be more than enough to adequately sample a child's phonology. A description of all of the PAL routines appears in the current issue of the Kansas University's Child Language Program's *Working Papers in Child Language*.

References

Dyson, A. (1987) The effect of phonological analysis procedure on the selection of potential remediation targets. *Language, Speech and Hearing Services in Schools*, **18**, 364–377.

Edwards, M. L. (1983) Issues in phonological assessment. *Seminars in Speech and Language*, **4**, 351–374.

Ferguson, C. A. and Farwell, C. B. (1975) Words and sounds in early language acquisition. *Language*, **51**, 419–439.

Goad, H. and Ingram, D. (1987) Individual variation and its relevance to a theory of phonological acquisition. *Journal of Child Language*, **14**, 419–432.

Griswold, Poage, and Polonsky (1968). *The Snobol4 Programming Language* (Englewood Cliffs, New Jersey: Prentice-Hall).

Grunwell, P. (1985) *Phonological Assessment of Child Speech (PACS)* (Windsor: NFER-Nelson; San Diego: College-Hill Press).

Hodson, B. and Paden, E. (1983) *Targeting Intelligible Speech: a Phonological Approach to Remediation* (San Diego: College-Hill Press).

INGRAM, D. (1981) *Procedures for the Phonological Analysis of Children's Language* (Baltimore, Maryland: University Park Press).
INGRAM, D. (1981/2) The emerging phonological system of an Italian–English bilingual child. *Journal of Italian Linguistics*, **2,** 95–113.
INGRAM, D. (1983) Jakobson revisited: some evidence from the acquisition of Polish phonology (Paper presented at the Annual Meeting of the Linguistic Society of America, Minneapolis, Minnesota).
INGRAM, D. (1986) Explanation and phonological remediation. *Child Language Teaching and Therapy*, **2,** 1–19.
INGRAM, D. (in press) *Child Language Acquisition: Method, Description, and Explanation* (Cambridge: Cambridge University Press).
KAHN, L. (1986) *Basics of Phonological Analysis: A Programmed Learning Text* (San Diego: College-Hill Press).
MILLER, J. and CHAPMAN, R. (1985) *SALT: Systematic Analysis of Language Transcripts* (Language Analysis Laboratory, Waisman Center on Mental Retardation and Human Development, University of Wisconsin-Madison).
PYE, C. (1987) *PAL: Pye Analysis of Language* (200 Arrowhead, Lawrence, Kansas).
PYE, C., INGRAM, D. and LIST, H. (1987) A comparison of initial consonant acquisition in English and Quiche. In K. Nelson and A. van Kleeck (Eds), *Children's Language*, vol. 6 (Hillsdale, NJ: Erlbaum).
PYE, C., WILCOX, K. and SIREN, K. (in press) Refining transcriptions: the significance of transcriber 'errors'. *Journal of Child Language*.
SHRIBERG, L. and KWIATKOWSKI, J. (1980) *Natural Process Analysis (NPA)* (New York: John Wiley).
WEINER, F. (1979) *Phonological Process Analysis* (Baltimore, Maryland: University Park Press).

Appendix A: Language sample entered in PAL format

$ Daniel, Father
\+ Age: 2;0(18).
\+ Context: Bookreading.
D hI:bVbgagi. (phonetic tier)
⟨ ⟩. (orthographic tier)
F What, honey?
F What's this here?
D 767 kHIpH. (each tier has a punctuation mark)
a tape.
F Taperecorder.
D kHipHukHOr6.
taperecorder. (words combined to show unproductive forms)
F Wanna go see Peter?
D kHey.
okay.
F You say it!
D nu nu. nu nu (a total of 4)
&no, &no, &no, no. ('&' indicates a repeated word)
F Oh.
F Aren't you gonna read the book?
D sebawu:kH.
⟨book⟩. (totally questionable utterance)
F Sit down!
D m::fb:.
⟨ ⟩. (totally unintelligible utterance)

D 7a7kHeytH.
okay.
F Okay.
F Here's Peter.
D pHiFu.
Peter.
F Yeah.
D eyyVvkix::.
⟨ ⟩.
F Let's see!
F What do we have here?
F What's this here, Dan?
D kHuA.
car.
F Right.
F Car.
D In 6 bowtH.
it a boat.
F mhm.
D E7 si:.
let/'s see. (use '/' to separate morphemes)
D n: nAnA.
a banana.
F Mm.
D sey nAnA?
see banana?
F Yeah, I see them, and they … (an incomplete utterance)
F What are they doing here?

Appendix B: Daniel's word index and phonological lexicon

Daniel's utterances:

1. hI:bVbgagi.
⟨ ⟩.
2. 767 kHIpH.
a tape.
3. kHipHukHOr6.
taperecorder.

*Number of Daniel's lines = 50
*Total number of lines = 106

Daniel's word index:

Word	*Freq*	*Sentences*
a:	5	2, 12, 14, ⟨22⟩, 23,
airplane:	1	18,
airplane/s:	1	21,
and:	2	22, 23,
another:	1	32,

banana:	2	14, 15,
basket:	1	50,
bike:	1	30,
boat:	3	12, 23, 29,
book:	4	⟨6⟩, 32, 33, 40,
call:	1	⟨45⟩,
car:	1	11,
corn:	2	48, ⟨49⟩,
dance/ing:	1	17,
it:	2	12, 45,
lace/s:	1	37,
let/'s:	1	13,
little:	1	29,
my:	1	33,
no:	4	5, 5, 5, 5,
not:	1	⟨33⟩,
okay:	5	4, 8, 26, 27, 31,
peter:	1	9,
pig:	2	42, 43,
plane:	2	19, 20,
recorder:	1	47,
see:	2	13, 15,
shoe:	2	34, 35,
tape:	1	2,
taperecorder:	3	3, 44, 46,
the:	1	50,
truck/s:	1	22,
want:	1	32,
what's:	1	45,

Daniel's phonological lexicon:

a	767 6 n: ⟨6⟩ 6
airplane	eytHeyn
airplane/s	ey7tE7nts
and	eynd 6d
another	n6nE
banana	nAnA nAnA
basket	bAsAtH
bike	BeykH

boat	bowtH bowt bowt
book	⟨sebawu:kH⟩ bEkH b6k bEkH
call	⟨kHow⟩
car	kHuA
corn	kown6 ⟨kHow⟩
dance/ing	dAnsin
it	In z6
lace/s	maysEn
let/'s	E7
little	Iy6
my	may
no	nu nu nu nu
not	⟨now⟩
okay	kHey 7a7kHeytH kHey kHey kHey
peter	pHiFu
pig	peytH peyg
plane	tHE:n hEn:
recorder	kHVd6
see	si: sey
shoe	su su
tape	kHIpH

taperecorder	kHipHukHOr6 kHeypHukHoyd6 kHikHowd6
the	d6
truck/s	kHats
want	n6
what's	h676

Appendix C: Daniel's abbreviated phonological lexicon

Orthographic form	*Phonetic target*	*Child's form*
a	6	6
airplane	eyrpleyn	eytHeyn
and	And	eynd
another	an6D&	n6nE
banana	b6nAn6	nAnA
basket	bAskHEt	bAsAtH
bike	baykH	BeykH
boat	bowt	bowt
book	bUk	bEkH
call	kHal	kHow
car	kHar	kHuA
corn	kHorn	kown6
dance/ing	dAnsiN	dAnsin
it	It	In
lace/s	lays6z	maysEn
let/'s	lEts	E7
little	lIF61	Iy6
my	may	may
no	now	nu
not	nat	now
okay	okHey	kHey
peter	pHiF&	pHiFu
pig	pHig	peytH
plane	plein	tHE:n
recorder	rikHord&	kHVd6
see	si:	si:
shoe	Su	su
tape	tHeyp	kHIpH
taperecorder	tHeyprikHord&	kHipHukHOr6
the	D6	d6
truck/s	trVks	kHats
want	want	n6
what's	wVts	h676

Appendix D: Daniel's phonetic analysis

Phonetic Inventory

p pHiFu peytH	t tHE:n	c	k kHow kHuA kown6 kHey kHVd6 kHIpH kHipHukHOr6 kHats
b bAsAtH bowt bEkH	d dAnsin d6	j	g
m maysEn may	n n6nE nAnA nu now n6	N	h h676
f	T	s si: su	S
v	D	z	Z
r	l	y	w
B BeykH	F	W	7

*
*
*

27 INITIAL CONSONANTS

i pHiFu si: kHipHukHOr6	I In kHIpH	e	E bEkH E7 tHE:n
A nAnA bAsAtH dAnsin	a kHats	O	o
U	u kHuA su	V kHVd6	6 6 n6nE d6 n6 h676

ay maysEn may	ey eytHeyn eynd BeykH kHey peytH	oy	6y
aw	ow bowt kHow kown6 now	a¯	i¯

*
*
*

31 VOWELS

p kHIpH	t bAsAtH bowt peytH kHats	c	k BeykH bEkH
b	d eynd	j	g
m	n eytHeyn eynd dAnsin In maysEn tHE:n	N	h
f	T	s kHats	S
v	D	z	Z
r	l	y	w
B	F	W	7 E7

*
*
*

16 FINAL CONSONANTS

Appendix E: Daniel's substitution analysis

Substitutions

p pHiF& pHiFu pHig peytH plein tHE:n	t tHeyp kHIpH tHeyprikHord&kHipHukHOr6 trVks kHats	c	k kHal kHow kHar kHua kHorn kown6

b b6nAn6 nAnA bAskHEt bAsAtH baykH BeykH bowt bowt bUk bEkH	d dAnsiN dAnsin	j	g
m may may	n now nu nat now	N	h
f	T	s si: si:	S Su su
v	D D6 d6	z	Z
r rikHord& kHVd6 trVks kHats	l lays6z maysEn 1Ets E7 1IF61 Iy6 plein tHE:n	y	w want n6 wVts h676
B	F	W	7

*
*
*

45 INITIAL CONSONANTS

i pHiF& pHiFu pHig peytH rikHord& kHVd6 si: si:	I It In 1IF61 Iy6	e plein tHE:n	E 1Ets E7
A And eynd bAskHEt bAsAtH dAnsiN dAnsin	a an6D& n6nE kHal kHow kHar kHuA nat now want n6	O	o kHorn kown6 okHey kHey
U bUk bEkH	u Su su	V trVks kHats wVts h676	6 6 6 b6nAn6 nAnA D6 d6
ay baykH BeykH lays6z maysEn may may	ey eyrpleyn eytHeyn tHeyp kHIpH tHeyprikHord&kHipHukHOr6	oy	6y
aw	ow bowt bowt now nu	a¯	i¯

```
*
*          33 VOWELS
*

p                    t                    c                    k
  tHeyp kHIpH          bAskHEt bAsAtH                            baykH BeykH
                       bowt bowt                                 bUk bEkH
                       It In                                     trVks kHats
                       lEts E7
                       nat now
                       want n6
                       wVts h676
b                    d                    j                    g
                       And eynd                                  pHig peytH
                       rikHord& kHVd6
                       tHeyprikHord&kHipHukHOr6
m                    n                    N                    h
                       eyrpleyn eytHeyn     dAnsiN dAnsin
                       And eynd
                       kHorn kown6
                       plein tHE:n
                       want n6
f                    T                    s                    S
                                            lEts E7
                                            trVks kHats
                                            wVts h676
v                    D                    z                    Z
                       an6D& n6nE           lays6z maysEn
r                    l                    y                    w
  kHar kHuA            kHal kHow
  kHorn kown6          lIF61 Iy6
  rikHord& kHVd6
  tHeyprikHord&kHipHukHOr6
B                    F                    W                    7
                       pHiF& pHiFu
*
*          33 FINAL CONSONANTS
```