

THE LOCALITY PHENOMENON AND PARALLEL PROCESSING OF
NATURAL LANGUAGE

E. L. Lozinskii and S. Nirenburg

Department of Computer Science
The Hebrew University
Jerusalem, Israel

1. Among the various traditions established in computer processing of natural language during the twenty-odd years of research the understanding that any such processing is to be done sequentially has a special status.

Even the most advanced natural language processing systems employ the sequential mode as a necessary evil, or do not even consider it an evil due to the ostensible lack of alternatives; thus, for instance, such well-known systems as SAM, PAM, ELI /cf. e.g. Schank and Riesbeck, 1981/, PHRAN /cf. e.g. Arens, 1981/ or PARSIFAL /see e.g. Marcus, 1979/ are all based on sequentiality.

The recent advances in the VLSI technology suggest that a re-evaluation of this tradition is in order. Indeed, non-sequential 'parallel' methods start emerging. In the field of AI one could mention, for example, Kornfeld's (1979, 1981) work in problem solving or the approach of HEARSAY-II (see Erman et al., 1980) to speech processing. The word parallelism seems even to turn gradually into a current 'buzz-word' in the AI community. Note that the meaning of this word still remains largely loose. Thus, Phillips and Hendler (1981) suggest a system of several task-oriented

processors working in parallel.

2. A different and a more powerful approach to parallel processing of natural language is suggested here: instead of functional distribution we suggest parallel distribution of input stream elements; a processor is assigned to every item of input and each such processor is provided with the same software package, so that all processes within a certain group become equal in status and modus operandi. (Note that this also increases the system's reliability, since even in the unlikely case of failure of $n-1$ processes the remaining one will accomplish the task by itself, in the sequential mode.)

Our approach to parallelism is based on the phenomenon of locality. Currently we apply it to constructing a syntactic parsing system for a subset of English, as a simple case of natural language processing.

3. Let us consider a text as a vector made up of discrete elements w_i :

$$T = /w_0, w_1, \dots, w_n/.$$

Being fed with T a certain Natural Language Processor (NLP) produces a structure of the form $S(T) = /v_0, v_1, \dots, v_m/$, where v_j can be of various nature: words in the object language and/or words and symbols in a metalanguage and/or various kinds of delimiters.

Let $D(v_j)$ be the minimal subset of T determining v_j in the sense that information carried in the elements of this subset is necessary and sufficient for outputting v_j by the NLP. Let g_j be the index of the leftmost element of $D(v_j)$ in the string T , and h_j , the index of the rightmost one (e.g. if $D(v_j) = /w_3, w_5, w_{10}/$ then $g_j = 3$ and $h_j = 10$). We now define the important notion of locality. Locality of an output element v_j is

$$l(v_j) = 1 - \frac{h_j - g_j}{h_j} = \frac{g_j}{h_j};$$

This function has a number of interesting properties. If an output element v_j depends on exactly one element w_i , then its locality is unity, the highest possible value. On the other hand, if a certain v_k depends on a large range of input elements, then its locality is close to zero. Comparing parallel and sequential processing we show that the ratio of the time necessary to produce an output element in parallel mode to that of the sequential mode strictly depends on the locality of this element. Moreover, the relative time gain of parallel processing as regards sequential processing is exactly the given element's locality. In other words, the greater the aggregate locality of elements in a certain text, the more benefit there is in its parallel processing. Such is the intrinsic connection between the notion of locality and the performance of a system based on parallel processing.

4. The process of implementation starts with finding clusters of high locality in the text. At this stage we prove the following

Proposition. NPs, VPs and PPs of English are highly local. On this basis we proceed to build a system of parallel parsing for English. In its present form the system consists of three modules: a morphological and two syntactic ones. The result of the first stage is a set of sets of dictionary entries for every input word, which determines the syntactic classes to which the input words may in principle belong. A grammar for each processor at the first syntactic stage of analysis is presented as a table which indicates all the correct triads of syntactic class members in the subset of English we are analyzing. This means that this stage is devoted to finding the states of compatibility between the 'neighbours' in the input string. It terminates when all the possible triads have been checked, produces candidates for correct parses (if any) and transfers them to the second syntactic stage whose task is a/ to carry out all kinds of agreement

and completeness tests (e.g. the subject-predicate number agreement and the presence of at least one verb in the sentence, resp.) and b/ to build one or more representations of the parse(s) (e.g. a constituent tree and a predicate-role structure). This modular framework facilitates the addition of new stages to the system, such as one or more semantic stages and an inferencing mechanism (provided a world is defined).

The communication between separate stages of analysis is accomplished globally, and here a secondary parallelism, this time the functional one (cf. Phillips and Hendler, op. cit.) can be implemented.

References.

- Arens, Y. (1981). Using language and context in the analysis of texts. Proceedings of the IJCAI-81. Vancouver, B.C.
- Erman, L. et al. (1980). The HEARSAY-II speech understanding system: integrating knowledge to resolve uncertainty. ACM Computing Reviews, 12, No. 2.
- Kornfeld, W. A. (1979). Using parallel processing for problem solving. MIT AI Lab Memo 561.
- Kornfeld, W. A. (1981). The use of parallelism to implement a heuristic search. Proceedings of the IJCAI-81. Vancouver, B.C.
- Marcus, M. (1979). A theory of syntactic recognition for natural language. MIT Press.
- Phillips, B. and J. Hendler (1981). Controlling parsing by passing messages. Proceedings of the Third Annual Conference of the Cognitive Science Society, Berkeley.
- Schank, R. C. and C. K. Riesbeck (1981). Inside computer understanding. Lawrence Erlbaum Associates, Hillsdale, N.J.