The number of 1s in vector V is equal to n (number of documents in the output), hence,

$$\sum_{i=1}^{N} (v_i)^2 = n.$$

The number of positions where 1 is contained simultaneously in K and V is equal to r (number of pertinent documents in the output), hence,

$$\sum_{i=1}^{N} (k_i \cdot v_i) = r.$$

Then

$$\cos \phi = \frac{\sum_{i=1}^{N} (k_i \cdot v_i)}{\sqrt{\sum_{i=1}^{N} (k_i)^2} \cdot \sqrt{\sum_{i=1}^{N} (v_i)^2}} = \frac{r}{\sqrt{c} \cdot \sqrt{n}}$$

$$= \sqrt{\frac{r^2}{c \cdot n}} = \sqrt{\frac{r}{c} \cdot \frac{r}{n}} = \sqrt{R \cdot P}.$$

Hence, the use of characteristic $\sqrt{R \cdot P}$ to evaluate the quality of the output is, indeed, justified.

Now we want to show that the use of $\sqrt{R \cdot P}$ to compare different outputs will avoid the computation of c. Let q_1 and q_2 be two query formulations (constructed from the same search request) with two corresponding outputs with recall and precision levels R_1, P_1 and R_2, P_2. Because both query formulations are based on the same request, the number of pertinent documents in the entire collection c is the same for searches based on q_1 and q_2. Hence,

$$R_i = \frac{r_i}{c},\ P_i = \frac{r_i}{n_i},\ R_i \cdot P_i = \frac{r_i^2}{c \cdot n_i},$$

where r_i is the number of pertinent documents and n_i is the number of documents in the i-th output ($i = 1, 2$). To determine which of the two outputs (corresponding to q_1 and q_2) is better, we need to compare the values $\sqrt{R_1 \cdot P_1}$ and $\sqrt{R_2 \cdot P_2}$ (the output is better if its $\sqrt{R \cdot P}$ value is larger). Hence, we are interested in finding the order of $\sqrt{R_1 \cdot P_1}$ and $\sqrt{R_2 \cdot P_2}$, that is $\sqrt{R_1 \cdot P_1} < \sqrt{R_2 \cdot P_2}$ or $\sqrt{R_1 \cdot P_1} > \sqrt{R_2 \cdot P_2}$. But the order of $\sqrt{R_1 \cdot P_1}$ and $\sqrt{R_2 \cdot P_2}$ is the same as the order of $R_1 \cdot P_1$ and $R_2 \cdot P_2$ and, hence, the same as the order of r_1^2/n_1 and r_2^2/n_2 since c is a nonnegative constant. Therefore, we conclude that to determine the order of $R_1 \cdot P_1$ and $R_2 \cdot P_2$, it is sufficient to consider r_1^2/n_1 and r_2^2/n_2 and, hence, in comparing the quality of different outputs we just need to compare their corresponding values of r^2/n. The criterion for the comparison of outputs (r^2/n) described here avoids a time-consuming user's evaluation of the documents in the collection. The only task required from the user is determining the pertinent documents in the output (parameter r), which is a normal (and natural) task in the interaction between the user and the system. Clearly, it is preferable for the user (rather than some intermediary) to determine which documents are pertinent. The user's evaluation of the output will be used by a feedback mechanism of a multiversion IR system for determining the best query formulation (or a combination of query formulations). Therefore, it is important for IR systems performing optimal search to have a feedback mechanism that is capable of evaluating different versions of query formulations and determining the best one (or best ones) among the existing versions. In the 1993 paper such feedback was called *selective feedback* because it realized the process of selecting the most appropriate state of the system (Frants et al., 1993).

We have discussed the properties of feedback for optimal search. Note that the methods and the algorithms for selective feedback also depend on how the collection of documents is used: statically or dynamically. Therefore, the feedback algorithms, described next, which consider the choice of an optimal alternative for the search, are oriented toward a specific collection type. The developed selection criterion r^2/n permits evaluation not only of query formulations but also of every subrequest in each query formulation. This is especially important for a dynamic collection of documents.

In describing the algorithms for selective feedback, we will assume that in an IR system realizing an optimal search the Boolean search criterion is used, and this system includes some set of algorithms for constructing query formulations in Boolean form. This set could, for example, consist of the algorithms proposed by Voiskunskii and Frants (1971), Dillon, Ulmscmeider, and Desper (1983), Salton, Buckley, and Fox (1983), Frants and Shapiro (1991a), and others. In each of these algorithms it is possible to use a marked set of documents as a search request, and each of the algorithms constructs a query formulation in Boolean form. For these reasons, all further discussion is applicable to the algorithms for constructing query formulations in IR systems realizing optimal search.

Before the initial search for a given search request, the query formulations are constructed by each of the available algorithms. The initial search is then conducted by each of the constructed query formulations and all outputs are combined. In other words, combined output will be formed as a set union of all the outputs obtained by each query formulation.

9.9

Selective Algorithm for the Static Collection of Documents

As was indicated earlier, the initial search is performed for each of these query formulations, and all of the outputs are combined into one output (dupli-