Working With "Project 1"
My experiences with computer composition

By computer composition I understand the formulation of sets of rules with the aid of a computer with a view to working out musical contexts without explicitly defining the acoustical presentation space. If an electronic studio is used for the acoustic definition, we get electronic music; if an orchestra is used, instrumental music; if nothing is used, it remains composition theory. (Sets of rules can also be used without resorting to a computer.) Hitherto, most of my experiments in the field of computer composition envisaged an instrumental presentation, performed by musicians. There is one exception: "Output", which I produced with the VOSIM system at the Institute of Sonology (Utrecht), controlled with "Project 1" data.

I was first moved to formulate sets of rules by serial composing methods, later by composing in the electronic music studio (sets of rules materialized in circuits and mechanical operations), and still later by studying a computer language and the need to practise programming. Although already adequately versed in systematic composition, I could not bring myself to compose a piece for the sole purpose of translating it into a computer language before writing the score. Instead, I cast my mind back to the early *Ferienkurse* in Darmstadt and the discussion about composition technique (Goeyvaerts, Stockhausen, Boulez); was it not time to put these techniques to a systematic test? Were there any reliable rules, and to what extent did they cater for the complexities of musical contexts, meaning acoustic (or graphic, if you like) data which make musical sense (whatever that may mean)? Does a composer actually "know" what he is doing – by which I mean: can he express his knowledge without applying it?

The only answer was to try it out. The outcome was "Project l", a Fortran program which described a generalized model of serial composition; it is currently installed at the Institute of Sonology in The Hague in an Atari version compiled by Ramón González-Arroyo [now, 2009, in a Windows-version, compiled by the author]. Generalization was largely a matter of renouncing the "original row plus permutation" constellation, which I replaced with random series (and hence random permutations) of data lists. Generalization was attended by limitations: the constraint that all functions described in the program should be used once in every run; the reduction to standardized list formats; the relinquishment of explicit durations within the context of time-intervals. This concept was midway between a generator for a large number of similarly organized compositions and a method for testing the program's strategy from one case to the next.

Formulating a strategy differs from composing a piece in that not details, but basic conditions, are established – in minute detail, however. Formulating a strategy, after all, means generalizing formal relationships, which is at variance with the common practice of expressing musical ideals as concrete musical forms. Generalizing, unlike specifying, means making sure that everything can occur once somewhere, but that it may not be "missing" either. It is hard to define "everything": as something both present and absent.

In order to realise this concept I recalled the technique of linking by means of overlapping. Music progresses in layers, so to speak, which can be considered as single parameters: rhythm, harmony, melody, dynamics, scoring. These layers run their own formal courses whose ends only coincide after longer formsections; the shorter the layers and hence the more unexpected their overlapping, the more clearly they articulate one another.

To distinguish several courses in a layer, I opted for a differentiation in degrees of static behaviour: from absolutely stationary (repetition) to resolution (the proper "serial" case) into as many steps as possible.

This strategy leads to a different concatenation of courses (or "groups') of different lengths in each layer; the shorter the groups, the more likely their coincidence in two or more layers, the more "short-winded", then, the formal progress; the longer the groups, the fewer coincidences, the longer the phrases. The strategy also results in more pronounced characterization when the layers are predominantly static or dynamic.

Details of this strategy are influenced by input data which are basically related to the time-layer: tempo and entry delays. Later versions of "Project 1" permit – as well as other additional facilities – the arbitrary combination of static degrees per section. In earlier versions this decision was left to chance; at the same time, though, the occurrence of the static degrees per layer was guaranteed throughout the piece.

The result of a program-run appears in the form of a table whose temporal succession of data is read from top to bottom; the layers ("parameters") appear in columns. Changing sequence of data, whether static ("groups") or dynamic ("rows"), are marked with asterisks, making the coincidence-pattern stand out. Some data refer directly to musical quantities (pitches, dynamic values), others only indirectly, calling for their interpretation according to context (instrument, register); still others have to be supplemented (rhythm: only the entry delays are given, durations being free).

This mix emerged fairly logically from my endeavours to formulate a compositional strategy which would afford neither complete freedom nor complete constraint. A hierarchy ranging from fixed, rhythmified pitches to a certain freedom of choice in the case of instruments and registers stimulates the idea of a rigid framework in which the parameters can interpret one another; on the one hand the missing durations leave the question of phrasing largely open, on the other hand they permit additional harmonic effects when long durations cause overlappings. Incidentally, harmony is indicated per time-point by a chord, the tones of which may be performed in succession, thereby exercising an extra influence on the rhythmic structure.

The interpretation of the score table serves the purpose of revealing the idea on which the input data are based; not the idea for a particular piece, perhaps, but for composition itself. When a single parameter is scrutinized, its characteristic structure emerges. If necessary, the composer could use this structure as the basis of a form-section by inventing the missing parameters. The same applies to the scrutiny of two columns which are mutually complementary and contradictory; both situations can be coped with by drawing on data from the parameters which are "free", i.e. unscrutinized as yet.

Things get more difficult with each further parameter, for although the parameters articulate one another by virtue of the underlying strategy, they also pin each other down. The composer is faced with the result of a process which he would otherwise (without a computer) have worked out and controlled spontaneously in phases. Control in "Project 1" is governed by the rules established in the program and activated by the composer's input data, which only invest those rules with meaning because they were borne in mind when the input data were formulated.

Spontaneous control is replaced here by "interpretation". Analytical scrutiny of the score table, framed by the original impulse on the one hand and the measures (the formulation of process data and process control) adopted to realize it on the other hand, by no means

precludes snap judgements. The interpretation of the score table is also an interpretation of the compositional strategy.

In interpreting the data structure composed by "Project 1", I favour a division of labour, a method, incidentally, with analogies in serial composing and electronic music.

Analysis of the score table starts with section-profiles: characterizations, section for section, yielded by the structures of the individual parameters. I am stimulated by predominant chord-sizes, loudnesses, registers or scoring indications in what are rather static fields, but also by the constant change of such quantities in more dynamic ones. It is therefore not necessary to sketch a profile of each parameter. When there is more than one parameter, the most suitable profile can be selected. I like to make a list of all the section-profiles in order to obtain a sequence of sections that will justify the overall form. This provides an idea of the "freedom" offered by the program strategy without violating the basic conception of the planned composition.

Within a section, the parameters can now be evaluated in the light of the profile in question. Here I distinguish between main and subsidiary parameters; what is a main and what is a subsidiary parameter depends on the respective section-profile. Main parameters are retrieved first and drafted on manuscript paper in a convenient fashion. This suggests details for the other parameters, causing me to wonder what decisions the program will have made for them. Now the subsidiary parameters are added to the existing draft, and my previous idea of the whole is modified. At each stage the data structure of the score table becomes clearer, more concrete; at the same time the original plan becomes more concrete too – the plan on behalf of which the process was set in motion.

If this method seems to be too "improvisatory" and hence to violate the data structure that is to be interpreted, the composer can try to evaluate the data in the table as literally as possible. The problem then is that the table structure, restricted to a few columns, is coarser than the imagined acoustic result suggested by the analysis. In such cases I devise a set of rules to be carried out by hand for each section. These rules generate the "fine structure" instead of leaving it to note-by-note evaluation, by imposing conditions under which the parameter data automatically modify one another, so to speak; general characteristics (such as static degrees), defining the static behaviour of the parameter data, also affect the rules. This method is used for larger instrumental ensembles, for instance, when justification for specifying individual instruments or groups of instruments has to be found in the data structure.

Dividing the labour in this manner has the advantage of allowing the composer to concentrate on single aspects of the composition (of composing) and their significance for the immediate context; nothing is lost, for the scaffolding of the composition was already erected in the score table and takes on more concrete form at every stage of the work.

Particular care must be taken in correlating the parameters when searching the score table for hints as to impending decisions. Scoring is only rudimentarily indicated in the score table, specified according to instrumental groups, for instance, not individual instruments. If you don't feel like improvising, you search the data in other columns for scoring suggestions. Such suggestions (which might even be concrete indications) are often to be found in characteristic parameter combinations which would otherwise remain uninterpreted it not brought to bear on decisions at various formal levels. It can thus come to pass that although the composition seems to be finished, the composer wonders whether any disregarded (uninterpreted) data combinations are left.

People often ask whether the computer replaces human spontaneity. The answer is that using the computer is the result of a spontaneous decision. The computer is assigned a place between the conception and the realization of a work. It creates distance, mental space; it relieves the mechanical drudgery of routine activity, as well as providing documentation.

The input data control a pre-structured process (the program's inherent strategy); correspondingly, the interpretation phase is orientated towards pre-structured data. The fabric is a fine weave; even where random decisions are made, all the data remain related to each other and to the compositional idea. Far from imposing a strategy on the composer, then, the computer becomes part of a higher strategy which unites the synthesis of the composition with the explicit analysis of composing.

The score table is the projection of the input data, as it were, seen through the lens of the compositional strategy, and is hence respected as a meaningful context; interpretation serves to preserve this context, not to obtain it from the data. Score tables which fail to reveal their relationship to input data or another superordinate criterium, are rejected out of hand.

Using computers has added a dimension to my musical experience. Admittedly, I have always been interested in "music in its technical rationality" (the title of a series of lectures which I gave in Bilthoven in the early sixties). After all, composing means fulfilling a musical desire, satisfying musical curiosity, getting to know music that doesn't (yet) exist. It leads to acts of objectification, realization, rationalization (= giving access to insight). Every score is a statement about music; "Project 1" is a collection of statements about music, couched in a language which is brought to utterance by mechanical means.

[1990]