

EVERYONE TALKS ABOUT THE WEATHER

**an installation for robot-controlled
pipe organ and weather satellite**

By Wayne Siegel, Professor

This is an account of my artistic research in working with a computer-controlled pipe organ. This project was carried out at the Royal Academy of Music in Aarhus between autumn 2012 and autumn 2013 and resulted in a 12-hour performance of Everyone Talks about the Weather, a sound installation for robot-controlled pipe organ and weather satellite presented at the closing of the Aarhus Festival on September 8th, 2013.

Klais organ, Aarhus Symphony Hall

Background

The pipe organ designed for and installed in the Aarhus Symphony Hall in 2010 was built by Klais Orgelbau in Bonn and contains over 3,000 pipes. Before its completion, I received a commission from the Aarhus Symphony with support from the Danish Arts Foundation to compose a work for the new organ with symphony orchestra. My Concerto for Organ and Symphony Orchestra was completed in 2011 and premiered in 2012 by the Aarhus Symphony with Professor Ulrik Spang-Hanssen as soloist.

While working on this concerto I had a chance to experiment with the instrument. As it turns out, the Klais organ in Aarhus is rather unique. The instrument employs direct electric action, allowing the organ to be controlled from two identical organ consoles in two locations: one fixed console situated near the pipes themselves, and another movable console, designed for placement on the stage. I discovered that the two consoles are connected by means of a built-in computer system. I have often worked with algorithmic composition systems that allow a computer to compose music in real time. I was curious to know if the instrument could be controlled by an external computer that could be programmed to play organ music without human intervention.

Artistic Vision

At first I was inspired by the idea of the computer playing music beyond what was physically possible for an organist to play. I imagined music that was fast, complex and that consisted of four independent “players” or voices. I did not stop at eliminating the

organist. Perhaps I felt guilty about making the organist obsolete. In any case I decided it was only fair to make myself, the composer, obsolete as well. My idea was to create a rule-based computer program that could compose and play music directly on the Klais organ. One machine (my computer) would be controlling another machine (the huge Klais organ). I was fascinated by this concept, but somehow it seemed that something was missing. After giving the idea some consideration, I decided that the work should in some way be controlled by an outside influence. I decided that the computer program should somehow be influenced by data from a weather satellite.

Technical challenges

First I had to get my computer to take control of the organ. Inside a small drawer on the organ console cabinet I found a small hardware device called the Holtzapfel MSD-1. This little box was designed as a MIDI (Musical Instrument Digital Instrument) recorder that would allow an organist to record a performance and later play it back on the organ itself. The MSD-1 was connected to the organ with an ethernet (IEEE 802) cable and the unit could be unplugged and moved from one console to the other. On the box itself were standard MIDI connectors (five-pin DIN) labeled “MIDI in”, “MIDI out” and “MIDI thru”.

I contacted Klais Orgelbau in Bonn and requested documentation for this device. Although the organ was built by Klais, the organ consoles and the MSD-1 were not. Klais had no documentation related to the organ consoles or the MSD-1. I was referred to two companies in Bavaria:

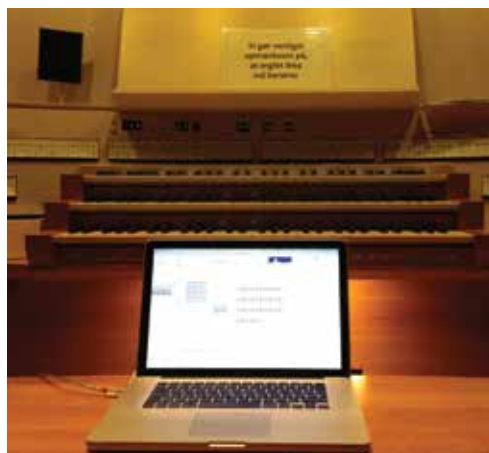
Eisenschmid, the organ console company that had designed and built the consoles and Holzapfel, a small engineering company that had designed and built the MSD-1. Each of the three companies referred to the other two for documentation, but in fact none of the three was able to provide any sort of documentation about the organ's MIDI implementation. I was forced to resort to experimentation. I connected the MSD-1 "MIDI out" connector to a standard MIDI interface connected to my computer and voila! MIDI note-on messages were being transmitted for each of the four manuals (choir, great, swell and pedals) on four different MIDI channels. When I connected the MIDI interface to "MIDI in" on the MSD-1 my computer could play any notes on the organ. I could now either send or receive MIDI note messages and my computer could control the organ.

The next step was to figure out how to control changes of organ stops or registration. After experimenting with changing registration I learned that registration changes were sent as MIDI system exclusive messages (sys-ex). The protocol used was more

complicated than I had imagined. I could record received sys-ex messages using the Max/MSP programming environment, but when I sent these same messages in various contexts back to the MSD-1 the changes of registration were not the same as the registration changes that I had originally received! After some bewilderment and head-scratching I decided that I needed professional help. My son, Gabriel, happens to be a computer scientist so I invited him to the concert hall and together we looked at the data stream received by my computer each time we changed a stop on the organ. When Gabriel saw the numbers flowing by he exclaimed: "it's a sum!" In fact the MSD-1 was transmitting several 8-bit binary numbers that together represented the sum of all of the registration switches added together every time a single switch was turned on or off. This meant that it was not enough to know the corresponding number of each of the switches that control the individual stops, it was also necessary to know the total state of all of the switches before a correct

Left. MSD-1 (shown here in the MSD-2 version)

Right. Computer controlling the Klais organ, Aarhus



registration change could be sent. After recognizing this technical hurdle I was able to begin writing software that would transmit sys-ex data corresponding to the total state of registration for all manuals whenever a single stop on any manual was turned on or off. Once this was accomplished, my computer was able to control all stops on the instrument.

There were still a few problems to be addressed. For fast passages (more than 8-10 notes per second) the organ had trouble keeping up and notes got clumped together in groups. Holzapfel was helpful in solving this problem. Also, occasionally data would get lost and a note would get stuck and not turn off. This problem was solved with a software work-around that turned all notes off at specified times.

Musical algorithms

Once I had successfully connected my computer to the organ I began developing software that could generate different modes of performance. This modular approach was inspired by Austrian composer Karlheinz Essl, who has worked extensively with algorithmic composition since the 1980's. Essl writes about his work *Lexikon Sonate* for computer-controlled piano (1992):

"Each module generates a specific characteristic musical output as a result of the compositional strategy that has been applied. A module represents an abstract model of a specific musical behaviour. It does not contain any pre-organized musical material, but a formal description of it and the methods how it is being processed."
(Source: <http://www.essl.at/works/>

Lexikon-Sonate.html#abstract).

I created eight different software modules that could play the organ in eight different performance modes. Each of the four organ manuals is controlled by its own version of one of these eight modules, allowing the four manuals to play independently of each other. The current performance mode used by any given manual is influenced by a global performance mode selector that changes performance mode every 45 seconds. The current tonality used by any given manual is influenced by a global tonality selector that changes tonality every 20-40 seconds. There are 28 different tonalities or keys that can be chosen. There are 24 different defined registrations, six for each of the four manuals corresponding to six different dynamics: pp, p, mp, mf, f and ff. The current registration used by any given manual is influenced by a registration selector that changes registration every 20 seconds.

Changes of performance mode, tonality and registration are not synchronized. On the contrary, each of the four manuals has its own built-in independence in the form of a random delay. For example, when the tonality selector changes to a certain tonality (e.g. from C minor to Ab minor), the four manuals will not change tonality immediately but wait various amounts of time from 0 to 18 seconds. This means that after a few seconds some of the manuals will probably start playing in the new tonality while others will probably still be playing in the old tonality. After a while the four "players" will tend to catch up with each other and play in the same tonality until the tonality selector again changes tonality. A similar process is used in changing registration (i.e.



Registration change page of the software

dynamics) and performance mode. In this manner, a complex and unpredictable relation between the four “players” arises. The four manuals will tend to play in the same mode, dynamic range and tonality but they will sometimes disagree about these three parameters. For all of these parameters, the software that I have created employs Markov chains to allow the computer to make decisions based on probabilities. Simply stated, Markov chains are used to define the probability of moving from one particular state to another, for example from one performance mode to another.

The eight different performance modes are described below.

Circular Melody creates solo melodies that gradually move from the lowest to the highest note on the keyboard. When the melody reaches the highest note on the current scale it begins to move gradually downward, when it reaches the lowest note in the current scale it begins to move gradually upward. The melody speeds up and slows down as it moves up and down the keyboard. The changing tempo is influenced by a random data generator and by data transmitted by the weather satellite. The

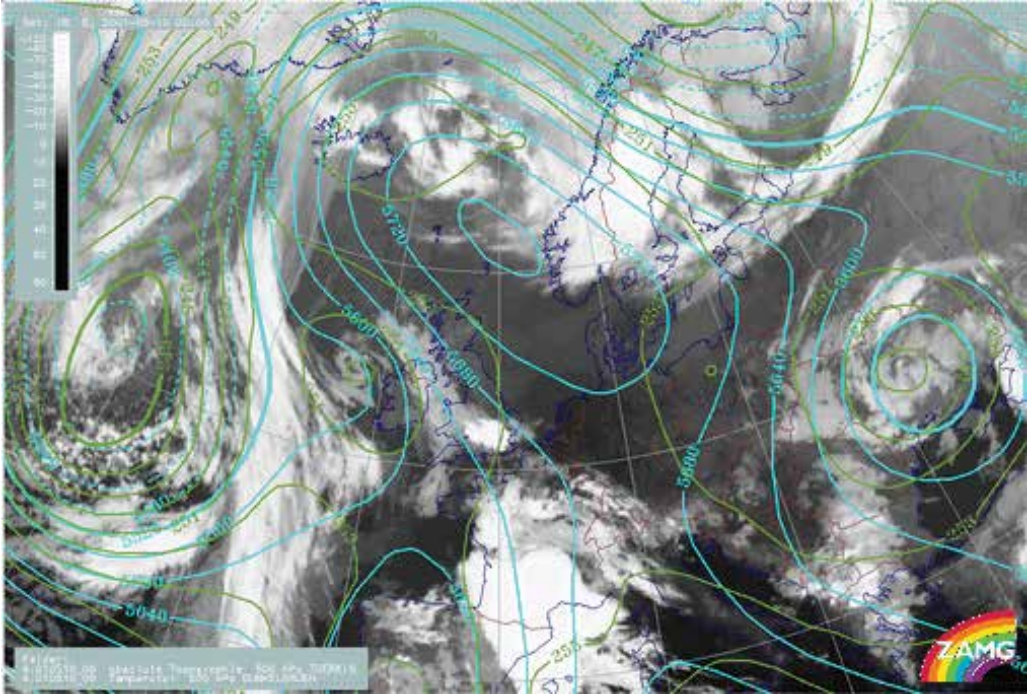
structure of each of the melodies is created using changing probabilities (again Markov chains) as to whether the melody will take one or two steps upward or downward. Changes of tonality are controlled by the global tonality selector as described above.

Meteor Shower plays accelerating and deceleration repeated notes. For each of the four manuals a hexachord of six notes within the current tonality is chosen. Each of the six notes is repeated independently with different maximum and minimum durations set for each of the six notes.

Trills is similar to Meteor Shower but creates six different trills based on a hexachord within the current tonality. The speed of each trill changes slightly over time.

Pedal tones also uses a hexachord within the selected tonality to play long sustained notes. The six notes selected do not start or stop at the same time.

Arpeggiator creates arpeggios within the selected tonality. The number of notes in the arpeggio changes constantly to create variation. Arpeggios for all manuals are synchronized.



Data image from Meteosat-10 fixed-orbit weather satellite

Riff Player creates rhythms and melodies based on set rules. The result is four independent voices that are rhythmically and tonally synchronized.

Chord Player is similar to Pedal Tones, but all six notes of the hexachord are played together (started and stopped at the same time) rather than separately.

Tacet is a kind of mute function in which a manual is silent. At certain times during the performance all four manuals might be in tacet performance mode, in which case there will be a general pause or silence.

Meteosat-10

In June, 2013 I contacted the Danish Meteorological Institute (DMI) in Copenhagen and told them about my project. My idea was to link some parameters of my compositional system to data transmitted by a weather satellite. Niels Hansen, the chief press officer at DMI, was extremely helpful and supportive. He suggested that I use data transmitted by Meteosat-10: a weather satellite in a stationary orbit above Europe. A dedicated ftp server was set up by DMI for the project. Weather data transmitted by Meteosat-10 was uploaded to the server every 15 minutes and this data could be downloaded directly by my computer program during performance. The data used in the piece is cloud-cover data from

192 locations in northern Europe ranging from the Czech Republic to Iceland. The weather data influences the overall tempi in some of the performance modules as well as the probability of various performance modes. Generally speaking, the influence of clouds will tend to make the performance more lively than if there are clear skies. In a sense, the weather is conducting the music while the computer is composing it.

Everyone Talks about the Weather

After creating the algorithmic software, linking my computer to the weather satellite and reworking and revising the compositional system I was ready to present the work for an audience.

Since my computer could compose and play endless hours of organ music without repeating itself, I decided to present a 12-hour concert, which would give the audience a chance to come and go and get a feeling for how the piece was progressing during a 12-hour period. The occasion for the premiere performance was the closing day of Aarhus Festival. Everyone Talks about the Weather was presented as a site-specific installation with no beginning or end. The audience was invited to enter and leave the hall at any time between 11:00 AM and 11:00 PM to experience how the composition (and the weather) developed in the course of 12 hours. The performance was full of unexpected events, even for me! Perhaps the most unexpected and magical part of the performance was the moment when a girl walked up onto the concert hall stage, sat down on the stage and after a few minutes started dancing!

BIO

Wayne Siegel was born in Los Angeles in 1953. Most important in the way of early musical influence was the American folk music tradition (Pete Seeger, Woody Guthrie, etc.). Major influences during the 1960's include the Afro-American blues tradition and avant-garde rock (The Mothers of Invention, Captain Beefheart). From 1971 to 1974 he studied composition and philosophy at the University of California at Santa Barbara (UCSB), where he concerned himself mainly with the European avant-garde tradition. After three years at UCSB he decided to complete his Bachelor of Arts degree while studying with Per Nørgård in Aarhus, Denmark. He remained in Aarhus and in 1977 he received his Danish degree in composition from the Royal Academy of Music in Aarhus. In 1978 he was awarded a three-year grant in composition from the Danish Art Council, working as a free-lance composer in the years that followed. After two years as administrative director of the West Jutland Symphony Orchestra and affiliated chamber ensemble, Esbjerg Ensemble, he was in 1986 appointed director of the newly founded national electronic music center, DIEM (The Danish Institute of Electroacoustic Music) in Aarhus. In 1994 he chaired the 19th International Computer Music Conference (ICMC) in Aarhus. From 1996 to 1998 he served as chairman of the two music committees of the Danish State Arts Foundation. In 2003 DIEM became part of The Royal Academy of Music in Aarhus, and Siegel was appointed professor of electronic music. In 2013 he was awarded a life-long artist's stipend from the Danish Ministry of Culture for outstanding artistic achievement.

For more biographical information read Waynes Biography Section @

http://waynesiegel.dk/?page_id=20



This article including illustrations and audio examples can be found here: www.waynesiegel.dk