Annales Geophysicae (2005) 23: 109–121
SRef-ID: 1432-0576/ag/2005-23-109
© European Geosciences Union 2005

Annales Geophysicae

Software Radar signal processing

T. Grydeland[1], **F. D. Lind**[2], **P. J. Erickson**[2], **and J. M. Holt**[2]

[1]Department of Physics, University of Tromsø, Tromsø, Norway
[2]MIT Haystack Observatory, Westford, Massachusetts, USA

Received: 5 October 2003 – Revised: 5 April 2004 – Accepted: 14 April 2004 – Published: 31 January 2005

Part of Special Issue "Eleventh International EISCAT Workshop"

Abstract. Software infrastructure is a growing part of modern radio science systems. As part of developing a generic infrastructure for implementing Software Radar systems, we have developed a set of reusable signal processing components. These components are generic software-based implementations for use on general purpose computing systems. The components allow for the implementation of signal processing chains for radio frequency signal reception, correlation-based data processing, and cross-correlation-based interferometry.

The components have been used to implement the signal processing necessary for incoherent scatter radar signal reception and processing as part of the latest version of the Millstone Hill Data Acquisition System (MIDAS-W). Several hardware realizations with varying capabilities have been created, and these have been used successfully with different radars. We discuss the signal processing components in detail, describe the software patterns in which they are used, and show example data from the Millstone Hill, EISCAT Svalbard, and SOUSY Svalbard radars.

Key words. Radio Science (Instruments and techniques; Signal processing; Interferometry)

1 Introduction

Since the initial demonstrations in the late 1950s (Bowles, 1958), incoherent scattering radars (ISRs) have proved to be very useful and one of the most powerful ground-based instruments for the exploration of the near-Earth space environment. Although the scattering cross section of the ionospheric plasma is extremely small at the typical frequencies used by these radars, the scattered power contains a wealth of information, making both detailed small-scale plasma physics investigations and large-scale geophysical studies possible. The radar's capability of observing very long time series and range-resolved information from a stationary geographical location complements the in situ measurements of rockets and satellites and the imaging potential of ground-based and space-borne optical instrumentation exceedingly well. This makes the ISR an invaluable instrument for building an understanding of space plasma physics and space weather phenomena.

The incoherent scatter radar technique has proved to be very useful. Accordingly, many different data acquisition and processing systems have been created over the last four decades (Hagen and Farley, 1973; Alker, 1979; Folkestad et al., 1983; Wannberg et al., 1997). These have been used to produce a significant database of geophysical measurements (Holt et al., 2002). Each of these systems has incorporated new capabilities and technological advances to improve the performance and capabilities of the radar systems with which they are used.

Recent advances in computing and software technology have recently allowed the demonstration of an ISR data acquisition and processing system, where the largest extent of the real-time processing is implemented in software using general purpose computers and networks (Holt et al., 2000). This system has now evolved from an early prototype into a production quality data system.

Radar implementations of this type are known as Software Radar systems and they have many similarities to Software Defined Radio systems which are being developed for communications applications (Mitola, 2000; Reed, 2002). The implementation of a production level incoherent scatter data system is a non-trivial task and a software focused system requires a significant information infrastructure to enable efficient and/or automated data processing and management. Important elements of this infrastructure are the signal processing components and the associated software patterns (Gamma et al., 1994) that describe the modules necessary for software-based down-conversion, digital filtering, and correlation based data processing.

The idea of a Software Radar is a very general one and has a wide application even beyond the examples discussed here. Many of the ideas which apply to monostatic active pulsed radars can be applied equally to other types of radar

Correspondence to: T. Grydeland:
(tom.grydeland@phys.uit.no)

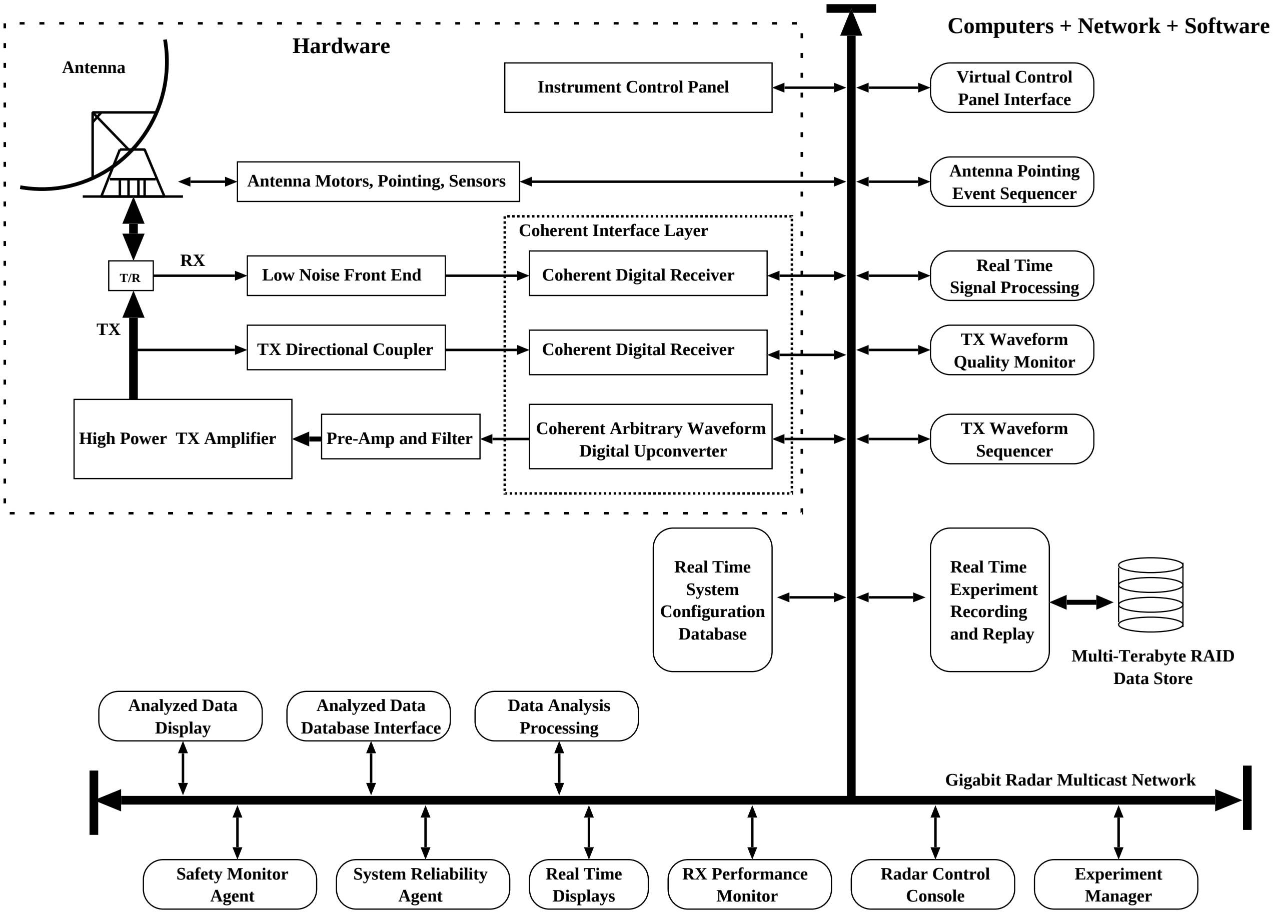
Hardware
Computers + Network + Software
Antenna
Instrument Control Panel
Virtual Control Panel Interface
Antenna Motors, Pointing, Sensors
Antenna Pointing Event Sequencer
Coherent Interface Layer
RX
T/R
Low Noise Front End
Coherent Digital Receiver
Real Time Signal Processing
TX
TX Directional Coupler
Coherent Digital Receiver
TX Waveform Quality Monitor
High Power TX Amplifier
Pre-Amp and Filter
Coherent Arbitrary Waveform Digital Upconverter
TX Waveform Sequencer
Real Time System Configuration Database
Real Time Experiment Recording and Replay
Multi-Terabyte RAID Data Store
Analyzed Data Display
Analyzed Data Database Interface
Data Analysis Processing
Gigabit Radar Multicast Network
Safety Monitor Agent
System Reliability Agent
Real Time Displays
RX Performance Monitor
Radar Control Console
Experiment Manager

Fig. 1. A simplified diagram of a generic Software Radar architecture. Analog elements are interfaced directly to a high speed multicast data network which provides the information transport necessary for real-time data processing. Multicasting allows many different software agents on a variety of computing platforms to share access to system data. High speed data recorders allow storage of command, status, signals, and radar output products.

architectures and even to more general examples of radio science instrumentation. For example, it should be possible to implement a distributed network of radio science instrumentation where the actual capabilities of the instrument are only realized in software running on powerful computing systems, potentially long after the data is collected. While the intent and behaviour of such a network might be quite different from a traditional monostatic radar system, the underlying Software Radar would employ many of the same patterns we describe here. Such an approach could also support novel organizational approaches, such as the sharing of computational resources between several instruments or dynamic reconfiguration in response to changing geophysical conditions.

2 Software Radar technology

A Software Radar system is a virtual instrument which exists in an information space and is connected to analog sensing elements via a coherent interface layer. This interface layer isolates the digital system from the reality of the analog world and ideally provides a high quality representation of selected radio frequency (RF) bandwidths for processing from a given electromagnetic sensor system. An example software radar architecture is shown in Fig. 1.

For Software Radar applications the coherent interface layer consists primarily of networked coherent digital receivers. Signals provided by the coherent interface layer must be sampled and transformed (with great linearity and phase stability) into the digital domain for radar applications. The traditional received radar signal (RX) which contains the return information from the remotely sensed region is the primary focus of the radar signal processing. This processing seeks to extract information about the target resolved in range, Doppler, and time. In the case of incoherent scatter, this is typically done by means of correlation functions which allow determination of physical ionospheric parameters through an inverse theory approach (e.g. Lehtinen and Huuskonen, 1996). In addition to the RX signal it is necessary for the software radar system to have knowledge of the waveform transmitted (TX) by the radar system. This knowledge can be implicit where the processing software knows what waveforms the radar intends to transmit at any

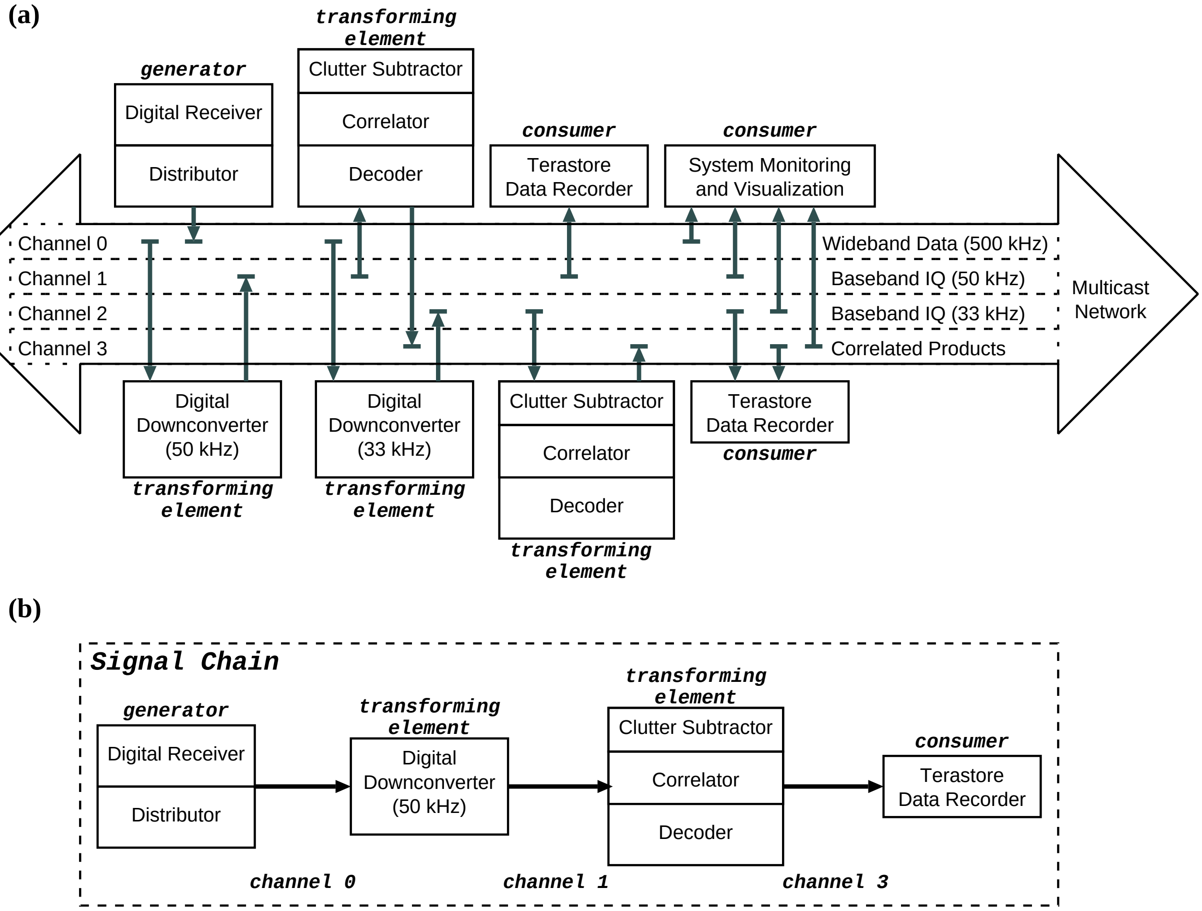

Fig. 2. Multicast channels provide a means of organizing information in a Software Radar system. By allowing multiple software agents to simultaneously access data in the system a high degree of parallelism is possible. (**a**) Data related channels and processing elements are shown at the network level. (**b**) A typical signal chain (cf. Sect. 3.1) used for producing correlated and decoded output as part of coded pulse processing. Multiple signal chains can exist using the same network channels and input data streams to provide different types of signal processing.

moment in time. Alternatively, it can be explicit where the TX waveform is digitized directly. In modern radar systems it is advantageous to digitize the TX waveform directly and to use the measurements of this waveform to aid the signal processing. This signal may be sampled prior to transmission by the radar hardware or intercepted in a manner similar to that used by passive radar systems (Sahr and Lind, 1997).

In practical terms an implementation of a Software Radar is an organized collection of software programs distributed among computing elements and connected by a high speed communications network. As in all distributed computing systems, there are limitations on the available computing power, data storage, and the ability to transport data between these elements. In particular, the underlying data network can impose significant limitations on the quantity of RF bandwidth transported, the latency with which processing occurs, and the reliability of the underlying data streams.

2.1 Multicast networks and signal processing

One possible implementation of a Software Radar system uses multicast communication (Stevens, 1998, ch. 19) over the data network to enable a high degree of parallelism in data management, processing, control, and system monitoring. For convenience, multicast traffic can be separated into multiple channels which may bridge physical networks, and which are allocated to individually handle a particular type of data traffic. A typical set of channels found in a Software Radar system includes one or more control, data, status, profiling, and debugging channels. Channels can be persistently or dynamically allocated depending on the particular requirements of the system.

An example of the channel organization associated with an incoherent scatter radar experiment is shown in Fig. 2. In this example, wide bandwidth receiver data (e.g. 500 kHz) is multicast onto a data channel (channel 0). This data is simultaneously accessed by separate processes which measure system noise temperature, digitally down-convert and filter the data, and display resulting products. The digital down-conversion processes each select a different received RF bandwidth (50 kHz and 33 kHz) and multicast the filtered baseband data onto two separate channels. The data produced on these channels is recorded separately on two different data storage systems, while clutter subtraction (covered in Sect. 3.4.4), and correlation and decoding (Sect. 3.4.5) are performed using other computing elements. This particular organizational structure is simply an example. A very large number of other possible configurations can be used as different processing needs arise and the configuration can be modified dynamically in response to changing processing requirements.

Because individual computing elements in a software radar system have finite processing and data management capacities, it is necessary to exploit multiple computing elements to meet the performance requirements of many modern radar applications. Networked information transport is necessary to enable this parallelism, and the use of a multicasting transport makes efficient use of network bandwidth to this end. Both coherent interface elements (receivers) and data processing elements are attached to the network and can communicate in a one to many fashion via multicasting.

The multicasting technique decouples the production of information (e.g. data sources) from its consumption (e.g. data processing, display, etc.), and lets any number of processes “listen” to the radar data streams. It also encourages a strong modularity in the data processing and software components. This is a key advantage, as it allows for highly scalable parallelism in the signal processing chains. Parallel compute-bound processing can be split between computers in both a simultaneous and pipelined fashion. Additionally, system monitoring tools and new experimental techniques can be tested and deployed side-by-side with the production implementations used for the regular operations of an instrument.

By encouraging, and in some cases enforcing, strong software system modularity, a multicast focused Software Radar architecture results in a signal processing system where each of the processing components performs a limited and well-defined operation in the radar system through a well-defined and network available interface. These properties encourage the development of generic components which can be reused easily and which are resistant to the occurrence of communication problems through the use of staged and threaded interfaces. Having clean interfaces between modules is also a great benefit during new component development, as modules can be tested and debugged in a controlled environment, and their operation can be verified either independently, or in parallel with the rest of the system.

2.2 Performance requirements and limitations

When considering the practical implementation of Software Radar systems it is important to understand the limitations on performance imposed by the networking and computing elements of the system. Modern networking components are available at a variety of performance levels ranging from tens of megabits to tens of gigabits per second in performance. In the future it is likely that even more capable systems will be developed and that the availability of high performance networks will become pervasive even over large distances.

Data in a Software Radar system is typically dominated by the raw sample streams from the A/D converters with a small percentage of overhead for framing and time stamp information. For incoherent scatter applications, sample sizes usually fit well in 16-bit words which allows a 100-Mbit network to carry at most 6.25 Msamples/sec or approximately 3 MHz of RF bandwidth. A gigabit network allows for a total of about 30 MHz of RF bandwidth to be transported. As the intrinsic bandwith of incoherent scatter ion spectra is typically less than 100 kHz at UHF center frequencies, this bandwidth is adequate for a significant number of such signals to be transported simultaneously. Also, modern network switches allow data rates of many tens of gigabits per second. By using switches which allow per port multicast routing (IGMP), it is possible to direct multicast traffic solely to interested parties, which reduces the networking overhead for each computing element.

One limitation of a Software Radar system implemented using a multicast data network is data transport reliability. The most common multicast transport implementation (using unreliable datagram protocol, UDP) does not guarantee that data which is transmitted onto the network will be successfully received. At first glance this would appear to be a serious problem, since the fundamental purpose of the data system is to acquire information, process it, and store it reliably. While there are reliable multicast transport mechanisms (Paul et al., 1997, and references therein), they have fairly high performance overhead, they have not been standardized, and they introduce significant complexity into the radar system software. Using a fully reliable transport (e.g. TCP/IP) is possible, but it removes the benefits associated with the multicast architecture almost entirely and, at the very least, uses bandwidth very inefficiently. Another solution to the reliability problem is to design tolerance for data loss into the processing modules instead of requiring fully reliable data transport. This is essentially a variation on the end-to-end principle for communications systems (Saltzer et al., 1984), a design concept which places a larger portion of the burden of communications reliability at the end points of a network system.

In most geophysical radar applications, data loss tolerance is the preferred approach for two reasons. First, under most conditions the actual rate at which data is lost once it has been transmitted is fairly low ($\sim$1%), so long as the computing elements which receive the data are not saturated. This loss rate can be minimized by careful system design and profiling of processing element performance. Second, most scientific radar applications are statistical in nature, and the occasional loss of a block of data is inconsequential to the estimation of radar derived parameters as long as the loss is detected. This is particularly true for incoherent scatter observations, which may integrate returned signals for several minutes in order to achieve sufficient statistical accuracy. Tolerance to data loss in the software introduces some complexity, but this is more than offset by the modularity, parallelism, performance, and debugging simplicity offered by the multicast approach. The resulting tolerance also helps in situations where data loss occurs for reasons unrelated to the data transport mechanism (e.g. intermittent receiver problems).

In cases where data loss cannot be tolerated, it is possible to combine a reliable transport with a multicast protocol to obtain the needed reliability at the expense of network bandwidth. For example, data from a coherent digital receiver could be multicast and recorded to a network file system drive simultaneously. Processing which required absolute reliability could act upon the data stored on the networked file system, while less critical processing could be performed on the multicast data. This approach does, however, increase the complexity of the resulting Software Radar architecture.

3 Signal processing software elements

The input data to the signal processing system of an active radar typically consists of contiguous blocks of samples which begin with the outgoing transmitter pulse and end after an appropriate receive interval. In the case of a Software Radar system, both a transmitter digitization and one or more received signal channels will typically be available for use in subsequent signal processing. This data can be organized into full interpulse periods (IPPs), or it can be a continuous stream of A/D samples which are organized dynamically as needed by different processing stages. A typical IPP is shown in Fig. 3 for a transmitter and receiver channel. Different sub-blocks within the IPP need to be processed in different ways by the radar system. For example, when the receiver protector is activated, the receiver channel samples can usually be ignored; ionospheric signatures will only exist over a finite set of delays (ranges); and a noise diode signal is often injected into the system for absolute system temperature calibration.

The primary signal processing related software pattern that has been identified in Software Radar systems is the Signal Chain. This fundamental pattern can be composed of a variety of other patterns including Generators, Selectors, Transforming Elements, and Consumers. These patterns are not specific to any programming language or even to the multicast architecture. They are fully generic organizational structures that will appear at least partially in any radar signal processing implementation.

The focus of this paper is on the Signal Chain pattern and patterns which are Transforming Elements appropriate for ionospheric radar applications. Other patterns, such as those needed for the management of data and control flow in a Software Radar, will be subject to more complete treatment in separate papers. Some of these patterns will be touched upon briefly here, such that the passing of data from source to finished products can be comprehended.

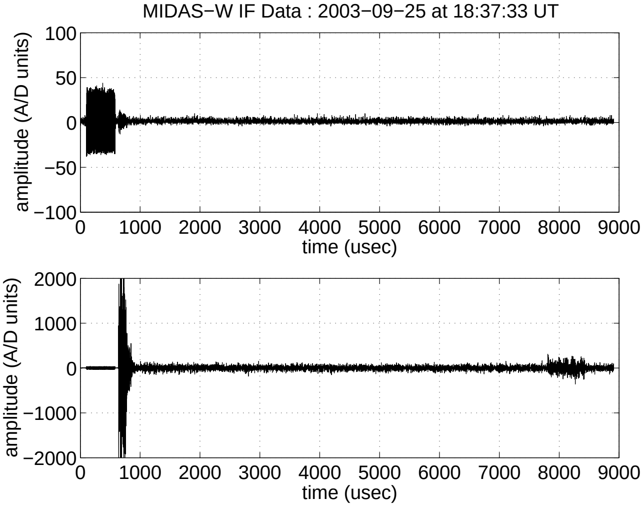

Fig. 3. An example of the source RF data (500 kHz BW) for signal processing from incoherent scatter observations made with the Millstone Hill ISR. Both transmitter digitization and received signal channel data are shown prior to digital filtering and downconversion. The outgoing radar signal is seen both directly in the TX channel (top) and as a leak through on the RX channel (bottom). Ground clutter and ionospheric returns are located in the main portion of the receive signal. A noise diode signal is injected for absolute system temperature calibration at the end of the sampling period.

3.1 Signal Chain pattern

The most common architectural pattern in Software Radar signal processing systems is the Signal Chain. The Signal Chain is an organizational pattern which captures the underlying structure of a signal processing system. A Signal Chain pattern may be embodied statically in a signal processing program or defined dynamically by a more generic implementation (e.g. signal processing chains defined in the eXtensible Markup Language, XML). The underlying pattern, its benefits, role, and consequences are independent of the particular implementation.

Signal Chains consist of one or more Generators that can be combined with a variety of Selectors and/or Transforming Elements. Each of the software components which form a Signal Chain performs a limited and well-defined task on input from a previous stage in the chain. The result of the processing is delivered to later stages in the chain and will ultimately be delivered to a number of Consumers that may monitor or record the resulting data products. Examples of Generators include an A/D sample distributor, numerical oscillators, and a variety of other signal producers. These signal producers can be arbitrarily complex, up to and including the generation of synthetic radar data for testing purposes. Examples of Selectors are satellite rejection (where anomalous data is discarded) or system failure mode detection (where normal data is discarded). Examples of Transforming Elements include several more traditional signal processing elements, such as digital filters, numerical mixers, Fourier transformers, and signal correlators. Consumers can record a stream of events to files, visualize data or system status, or simply indicate whether a particular channel has activity at all. By combining sequences of these simple elements, it is possible to build complex signal processing systems which are capable of the full range of radar data processing.

Signal Chains operate in an input to output cascade that is limited in performance by the slowest processing stage in the system. The stages in the chain can be individually threaded to allow for local parallelism, and tightly coupled stages, e.g. mean subtraction over a sample vector and subsequent filtering/decimation can be connected using asynchronous buffers. When used in conjunction with a multicast network, the resulting signal processing can exploit a collection of parallel computing elements (e.g. a traditional cluster computer) to increase processing performance. It can also use the modularity of the multicast architecture to hide custom signal processing hardware behind a network interface.

One important requirement of the Signal Chain pattern is "type" control of inputs and outputs. Elements of a chain allow only certain data "types" as inputs and produce only certain "types" as outputs. Thus, the elements of the chain are not arbitrarily configurable without the use of adaptive layers to convert and filter portions of the processed data stream. Individual elements of the signal chain may also require the synchronous provision of information to allow processing to occur. An example of this is a cycle dependent correlator which requires knowledge of the overall radar schedule and a particular data block's relation to it in order to correlate correctly. The provision of synchronous information in this context is known as a weaving operation.

A major difficulty in implementing Signal Chain patterns can be the asynchronous nature of communications in the system. This is particularly true for chains which are fully distributed across multiple computing elements using a network. This difficulty most often arises in the form of flow control problems which allow one element of the chain to saturate or fall behind while others are starved for data. So long as communications within the chain are reliable this is not an insurmountable problem because processing will slow to the rate of the slowest element in the system. However, when unreliable communications (e.g. multicast) are used, it is necessary to couple the Signal Chain pattern to other control and communications patterns, in order to provide flow control throughout the chain and load balancing. This process can be facilitated by providing profiling information at the inputs and outputs of the individual Signal Chain elements.

3.2 Generators

Generators are a pattern which occurs as sub-portions of a Signal Chain and which produce information for use in later stages of the signal processing system. The most basic element of this type is the distributor which obtains data from an external source and formats this data for use in the Signal Chain.

Another common type of occurrence is the proxy distributor, which can be used to generate synthetic signals in the radar system for operation of particular signal processing elements or for testing subsystems with known data, e.g. data with a fully predictable result or from a previous experiment with known results.

As part of our Software Radar implementation we have constructed a numerical oscillator component. This component is configurable to provide coherent oscillators in the signal processing system with programmable phase noise and spurious-free dynamic range levels. This allows the precision and memory usage of the oscillator representation to be traded off against the RF performance requirements of the signal processing system. The oscillator is capable of generating arbitrary waveforms at a given frequency and can also produce a linear chirp signal. Phase coherence in the oscillator can be tracked and maintained for use in coherent radar processing techniques. This oscillator component is most often used in conjunction with a numerical mixer for digital down-conversion of RF signals prior to filtering.

3.3 Selectors

Selectors are a pattern which allows some portions of an input data stream to pass through untouched, while other portions are blocked. Realizations of this pattern serve to determine the information which a particular portion of a signal chain will process and to isolate later processing stages from decisions relating to input validity. A basic example of this is a processing chain which only processes a particular type of radar waveform. For example, a particular signal chain may only be used to process coded radar data and this module will need to determine which of its inputs pass to later processing stages. Another example is a radio frequency interference (RFI) and/or satellite echo rejector, which blocks data based on derived metrics, such as long coherence times, or on a priori information provided by external agents (e.g. adjacent frequency monitors or externally supplied satellite ephemera). Another realization of this pattern might identify error conditions occurring within other parts of a Signal Chain, such as A/D sample distribution malfunctions in a distributor, and discard all resulting corrupted data until the fault has been cleared. This functionality is important to maintain system reliability and tolerance to abnormal conditions.

3.4 Transforming Elements

Transforming Elements are a signal processing pattern which transforms input data into output data as a stage in a Signal Chain. Implementations of this pattern do most of the work in the signal processing system and are usually the most intensive portions of a Software Radar. In many cases it is necessary to combine individual Transforming Elements into a single processing stage in order to achieve high performance. This optimization typically comes at the expense of flexibility and parallelism in the processing system. Example

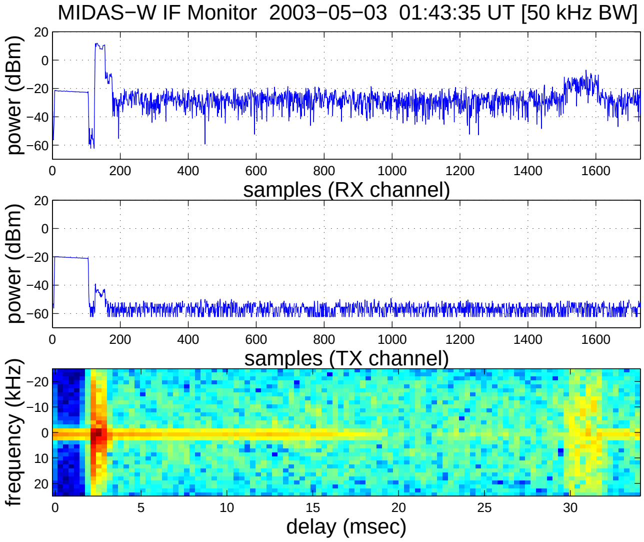

Fig. 4. An example of digitally down-converted and filtered data from the Millstone Hill radar system. Here intermediate frequency (IF) data at 500 kHz of bandwidth has been down-converted and filtered in real time to 50 kHz bandwidth data. The resulting 50 kHz bandwidth data is shown for the main receive (RX) and transmitter digitization (TX) channels. The bottom panel shows a spectral analysis of the RX signal.

realizations of this pattern useful in radar signal processing follow.

3.4.1 Numerical mixers

We have created a numerical mixer component which mixes the data from a numerical oscillator with an input sample stream. The mixer can handle real or complex input signals and can use a fixed mixing vector for speed where phase coherence between processing blocks is not required.

A typical use of this processing component is as part of a digital down-conversion Signal Chain. Digital down-conversion is necessary for a signal which is sampled away from baseband. This often occurs with RF and intermediate frequency (IF) sub-sampling receiver systems where channel selection and baseband IQ signal generation is performed in software.

3.4.2 Digital filtering and decimation

We have created a digital filtering and decimation component based on a finite impulse response (FIR) filtering approach for use as a general purpose digital filter. The digital filter provides for a programmable number of taps and coefficients and can perform any decimation which is an integer ratio of the sampling rate on the resulting signal stream. Approaches using infinite impulse response (IIR) filters are equally straightforward to include.

A common task for this component is in channel selection as part of a digital down-converter. Figure 4 shows an example of digitally down-converted and filtered data from the Millstone Hill incoherent scatter radar. This data has been processed by the baseband converter Signal Chain which operates in real time to reduce wide bandwidth RF signals (500 kHz) to narrow band RF channels (50 kHz), to reduce the load on subsequent processing stages.

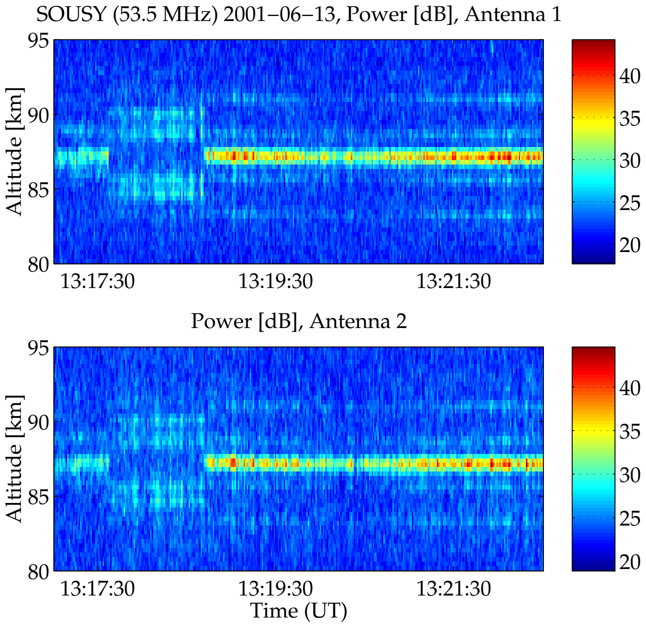

Fig. 5. Down-conversion and filtering is sufficient to produce some traditional radar data products, such as Range Time Intensity plots. Here data from an experiment using the SOUSY Svalbard radar taken using a MIDAS derived software radar system is shown. A PMSE layer is visible in the data. The secondary layering seen is most likely an artifact due to a mismatch between the sampling rate of the data system and the baud length of the complementary code sent by the transmitter.

Digital filtering can also be used to create a variety of radar output products. In Fig. 5 data from the SOUSY Svalbard coherent scatter radar at 53.5 MHz (Czechowsky et al., 1998) is processed by digital down-conversion and filtering to produce a range time intensity (RTI) diagram. Of course, due care should be taken when configuring radar processing signal chains because it is always possible to produce data that is difficult or impossible to process for some applications. For example, in the SOUSY data example here, the sampling rate converted to for the data is significantly different from the underlying complementary code baud length. This results in artifacts in the data which are unrelated to the geophysical returns.

3.4.3 Background subtraction

For certain radar experiment configurations (e.g. long pulses used with a monostatic single polarization receiver), the measured correlation functions contain contributions not only from the signals of interest but also from cosmic noise and receiver noise (Farley, 1969). These latter contributions must be removed to obtain the signal correlation function alone.

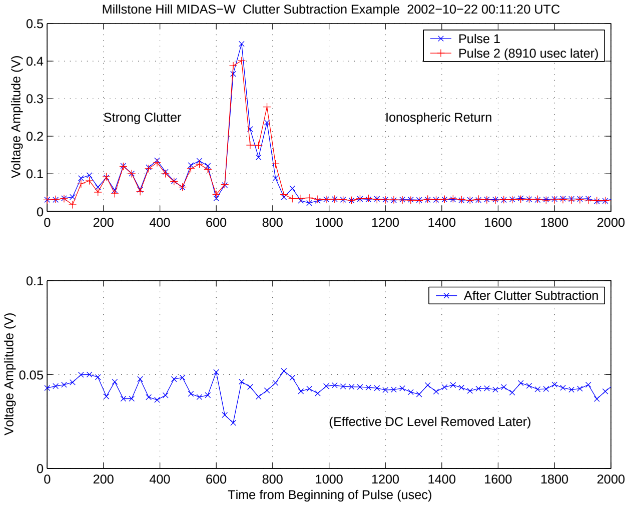

Fig. 6. A typical example of the clutter subtraction process from the Millstone Hill Software Radar implementation. The top plot shows the voltage amplitude of two successive pulses with identical transmitted waveforms from an alternating code sequence, containing strong clutter contamination up to delays corresponding to E-region altitudes. The bottom plot shows the result of modified two-pulse clutter subtraction (note the amplitude scale change), demonstrating short range clutter suppression.

(Other modulations such as alternating codes do not need this step, as self-cancellation of noise contributions occurs within the decoding operation, cf. Lehtinen and Häggström, 1987.) We have implemented a Transforming Element within our incoherent scatter Signal Chain which performs this background subtraction by separately estimating the noise correlation function at long ranges when the transmitted signal has become negligible, and subtracting this estimate from the result for the signal plus noise correlation. An alternate algorithm measures noise correlation by occasionally turning off pulse transmission (thus momentarily interrupting the experiment) and using samples taken during these periods.

3.4.4 Clutter subtraction

Radar clutter, uncorrelated signal from unwanted ranges or targets, arises both due to ground reflections and reflections from individual ground or air targets, and can be a severely limiting issue when measuring ionospheric correlation functions whose mean range is coincident with the unwanted clutter's apparent range. Suppression of the clutter signal, which can be more than 20 to 30 dB above the weak ionospheric return, is therefore crucial to obtaining meaningful results. We have implemented a Transforming Element which performs a modified version (to be detailed in a future publication) of the two-pulse clutter subtraction scheme described by Turunen et al. (2000). The module requires that each individual radar waveform be transmitted twice in succession, and the algorithm then performs voltage level clutter subtraction to produce a clutter-free IPP worth of receiver channel samples which can be processed by later stages to extract correlation information on the ionospheric signal alone. An example from the Millstone Hill radar is shown in Fig. 6.

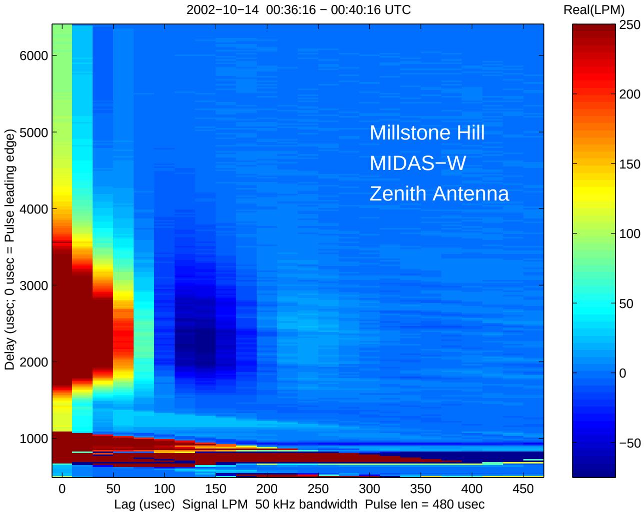

Fig. 7. A typical example of LPM level information from the Millstone Hill Software Radar implementation. Here, data is shown from the real portion of the LPM for a four-minute integration using a 480 μs pulse. Lags are computed at 20 μs intervals.

3.4.5 Correlation estimation

In both incoherent and coherent scatter radar, the details of the spectral power distribution of the scattering process is of interest, and for incoherent scatter radar, the usual way of obtaining the spectral information is by way of the autocorrelation function (ACF) of the scattered signal (Farley, 1969).

As discussed by Turunen (1986), and for alternating codes by Wannberg (1993), the best way of forming integrated correlation estimates for most incoherent scatter applications is through a lag profile matrix (LPM), called UNIPROG (UNIversal PROGram) in the earlier references. The complex baseband sample vector is multiplied with its own complex conjugate at all desired lag increments. The resulting lagged products are independent estimators of the same underlying physical process, and they can be integrated (summed). An example of incoherent scatter derived lag profile matrix is shown in Fig. 7.

If desired, ACF estimates can then be formed by making suitable sums along the diagonals of this matrix for groups of lagged products with the same or similar spatial contributions (Turunen and Silén, 1984). For long pulse modulations, the optimal summation is called trapezoidal summation (Holt et al., 1992). Such summation discards available spatial resolution, and for optimal analysis (Holt et al., 1992; Lehtinen et al., 1996), an entire LPM should be fitted at once. Figure 8 shows an example of ACFs formed from Millstone Hill radar data along with the corresponding spectra.

LPM computations are useful for all currently employed incoherent scatter techniques for ionospheric observations, and in most cases, only a small amount of post-processing

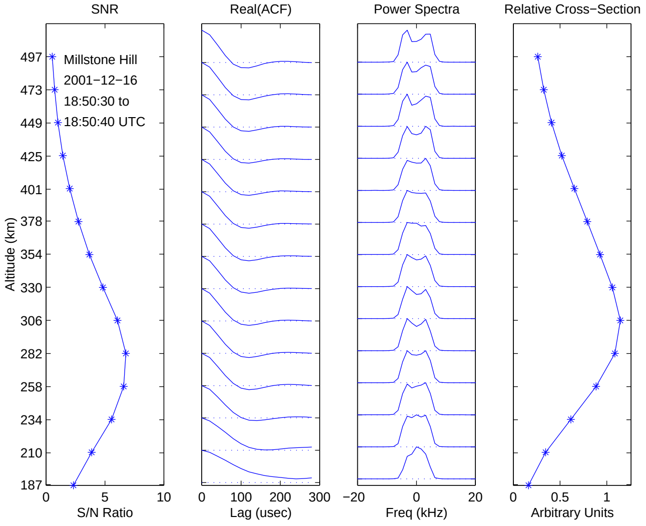

Fig. 8. An example of range dependent incoherent scatter correlation functions and spectral returns derived from a lag profile matrix using the Millstone Hill Software Radar implementation.

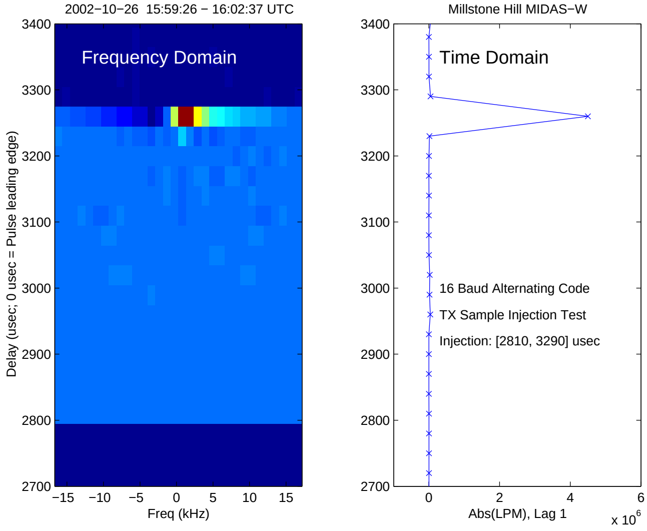

Fig. 9. Data from the outgoing transmitter digitization channel can be inserted into the receive samples at a known location to verify correct decoding of alternating code data. This figure shows the result of the processing, versus range (delay) and frequency in the left panel, and versus range (time) in the right panel.

separates the different modulations from a signal processing point of view. For coded pulses, LPMs are formed separately for each modulation. At the end of integration, all intermediate LPMs are decoded and summed to a result LPM. The decoding stage consists of applying a (±1) FIR filter generated from the code along each lag profile (Huuskonen et al., 1996). For random codes (Sulzer, 1986) and Lehtinen-type alternating codes (Lehtinen and Häggström, 1987) (but not for the alternating codes discovered by (Sulzer, 1993)), every decoded lagged product is ambiguity-free, so partial gates with fewer lags (and worse statistics) can be computed before and after the first and last complete gate. In practice, only the partial gates before the first complete gate are of interest.

When the alternating codes are oversampled, lag profiles can be computed at lag values which are not an integer multiple of the baud length. These lag profiles can be decoded in two different ways, a technique called "fractional lags" by Huuskonen et al. (1996). The resulting lag profiles have slightly different range contributions, so for optimal analysis they should be considered separately.

In multi-pulse codes (Farley, 1972) and alternating codes (all flavours), the zero-lag profile is ambiguous in range, and has traditionally been discarded or been used only for channel balancing (Turunen and Silén, 1984). As discussed by Lehtinen and Huuskonen (1986), this ambiguous data can be useful when inversion methods are employed, so in our implementation, the zero lag profile is always computed. For power profiles, an LPM with a single (zero) lag is used.

We have written a parametrized LPM module which is capable of computing the LPMs for all of these modulations, complete with methods of performing the post-processing for the decoding of finite sequence random, alternating codes, and fractional alternating codes. Methods are also included for the extraction of an ACF matrix from the LPM using several commonly used summation rules. Some processing capabilities have not yet been implemented. For example, multi-pulse sequences, arbitrary random codes, and aperiodic codes (Uppala and Sahr, 1996) will be supported in future software versions.

As part of validating the alternating code processing for correctness we have also implemented the capability of inserting a copy of the transmitter channel waveform into a desired part of the receive voltage samples. This is useful for determining the absolute performance level of coded pulses. An example of this transmitter waveform insertion is shown in Fig. 9.

For incoherent scatter radar applications it is necessary to process the autocorrelation functions using an appropriate fitter and resolve estimates of physical parameters with appropriate variances. Figure 10 shows the ionospheric measurements resulting from a full day of long pulse incoherent scatter data taken with the Millstone Hill radar and processed using the signal processing elements described above, while Fig. 11 contains the corresponding day of alternating code data which have undergone additional clutter subtraction and decoding stages.

3.4.6 Cross-correlation estimation

When multiple receiving antennas are available, interferometric techniques can be employed (Farley et al., 1981). In this and related publications, (cf. Kudeki and Farley, 1989) direct Fourier transforms over short sub-blocks of voltage samples were used to estimate the power spectra and cross spectra of the scatterer, as these are the quantities necessary for coherence estimation.

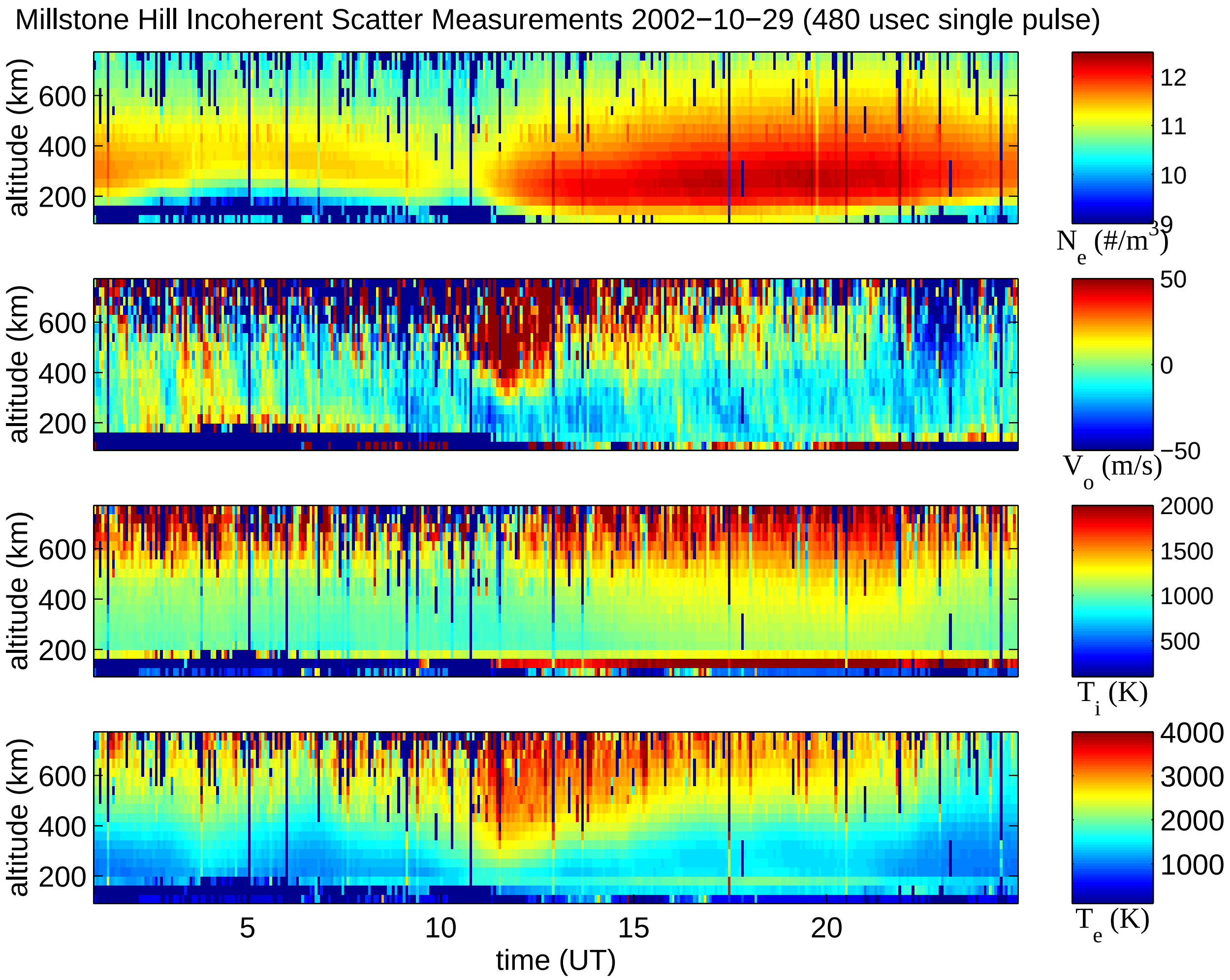

Fig. 10. Fitted results from a day of single pulse data at Millstone Hill processed using the Software Radar system. Electron density (log), Doppler velocity, ion temperature, and electron temperature are shown over a period of 24 hours. Four-minute integrations are used in this example but the data can be reprocessed from the voltage level to produce any integration time required and statistically acceptable.

As mentioned above, the optimal range resolution when estimating power spectra for long pulse modulations is obtained by way of the autocorrelation function using trapezoidal rule summation. For interferometry, we have extended this technique to the formation of cross-correlation function (XCF) estimates. These are analogous to autocorrelation function estimates, but with some essential differences. First, there are two inputs instead of one, so the symmetry which lets us compute only positive lags in autocorrelation function estimates no longer holds, and we have to compute both positive and negative lags. Second, since noise in the two inputs is uncorrelated, there is no need for background subtraction. Third, each input's ACFs must be computed individually for normalization. Range gating for cross-correlation function estimation using trapezoidal rule summation is illustrated by the products marked with black circles in Fig. 12, corresponding to Fig. 3 by Grydeland et al. (2004). For interferometry with alternating codes, intermediate LPMs with contributions from a single range are indicated by open and shaded circles in the same figure. During decoding, the contribution from all other ranges is eliminated, and the resulting lag estimates end up in the positions indicated by the shaded circles. A description of the interferometric technique, including estimation of scattering structure size is found in Grydeland et al. (2004).

We have implemented intermediate and resulting LPM computations for cross-correlation function estimation (XLPMs) in the signal processing module. These were used for the processing of interferometric observations of naturally enhanced ion-acoustic echoes at the EISCAT Svalbard Radar (Grydeland et al., 2003, 2004). An example of interferometric results from the EISCAT Svalbard Radar system is shown in Fig. 13.

3.5 Consumers

Signal Chains terminate in Consumers which accept data and perform functions which do not require further signal processing by a later stage. The most basic realization of this

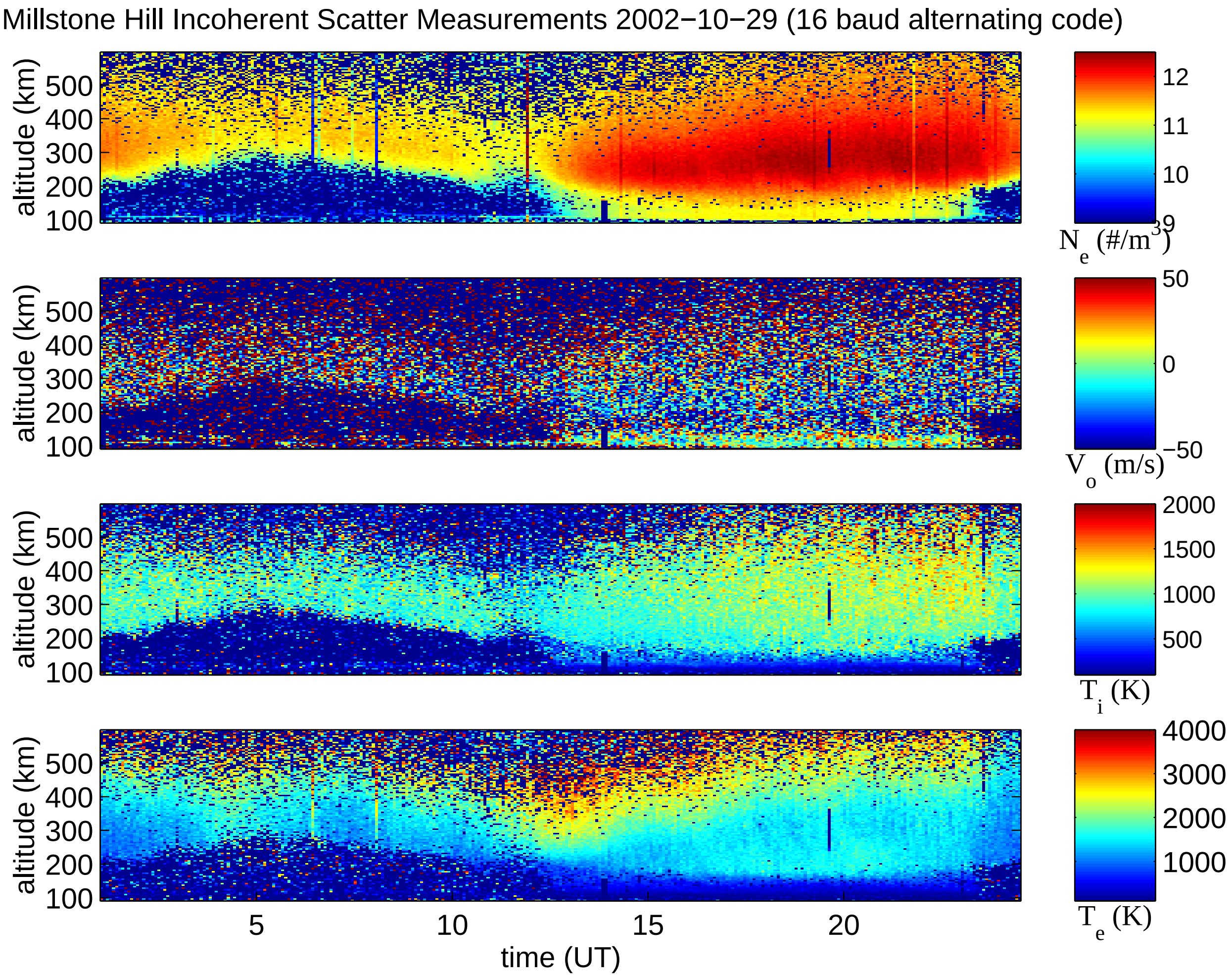

Fig. 11. An example of Millstone Hill alternating code data processed and fitted covering the E- and F-regions of the ionosphere over the course of a day. Here a sixteen-baud strong alternating code is used with a 30 μs baud and an overall pulse length of 480 μs. This code was interleaved with the corresponding single pulse during radar operations to allow for different measurements with simultaneous averaging centers to be made. Clutter subtraction was enabled for the processing, resulting in the recovery of E-region data. To allow for clutter subtraction, each alternating code sequence was sent twice and the return signals were subtracted at the voltage level.

pattern is a data recorder which transfers the end product of a Signal Chain to a more permanent storage, either in a fixed location, such as a disk file, or to another information domain such as a separate network. Other examples are data visualization agents, which present text-based or graphical summaries of outputs in either quick-look or publication quality format. For our Software Radar, we have implemented several Consumers, including a multicast channel activity monitor, a channel recorder, a real-time voltage level signal monitor, and an RTI plotter.

4 Conclusions

We have implemented a signal processing system for incoherent scatter radar data processing using a distributed multicasting Software Radar architecture. We have applied the signal processing components to a number of significantly different radar systems. Through these applications we have demonstrated the capabilities of our software for implementing the full range of current processing necessary for modern scientific radar observations. In particular, we have implemented a full production quality incoherent scatter processing system capable of coded and uncoded pulse processing and we have demonstrated interferometric observations using cross-correlation techniques.

As part of the development process we have identified key software patterns which exist in radar data processing systems. These patterns include Signal Chains, Generators, Selectors, Transforming Elements, and Consumers, which are combined architecturally to implement particular signal processing structures, both within a single program and in a distributed data processing system.

For our particular Software Radar implementation we have used a multicasting architecture to enable a high degree of parallelism in the data processing system. This design choice has resulted in a modular and scalable ISR data system which

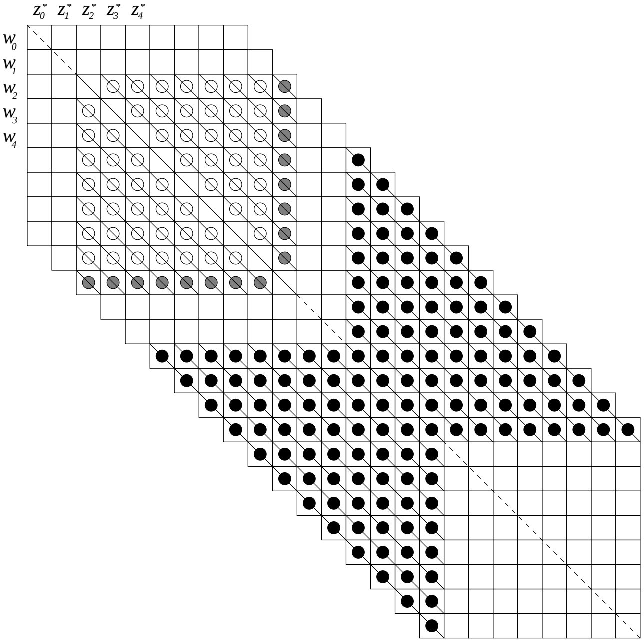

Fig. 12. A small cross-correlation lag profile matrix (XLPM), illustrating the shape of a range gate for coded and uncoded modulations, where points considered together (a single lag) are connected with diagonal lines. For alternating codes, the open and shaded circles constitute a range gate. In the decoded XLPM, the shaded circle indicates the position of the points resulting from this range gate. For fractional alternating codes, the picture will be slightly more complicated in the decoded XLPM, as each non-integer lag can be decoded in two ways. The black circles indicate a range gate for a long pulse XCF estimation, where the trapezoidal summation rule commonly used for long pulse ACF estimation is extended to negative lags. This part of the figure corresponds to Fig. 3 by Grydeland et al. (2004). In this case, four points are added for the zero lag, five for the first lag, etc. Seventeen lags are computed in total: eight positive, eight negative and the zero lag.

is also adaptable to new techniques and a variety of radio science applications.

We will make our Software Radar implementation available freely via the Open Radar Initiative (www.openradar.org). The Open Radar Initiative is an open source project supporting technology development for radio science applications. Refinement, documentation, and modularization of our Software Radar signal processing implementation will continue to improve as an overall system and an underlying collection of processing components. The resulting ISR data processing system should remain useful for significantly longer than hardware focused radar implementations.

Acknowledgements. The authors would like to thank J. Sahr for helpful discussions on many of these topics. The experiment at SOUSY Svalbard Radar was conducted by C. La Hoz and S. Saito. Thanks to S. Saito for providing the data for the SOUSY figure.

This work was supported by the EISCAT Scientific Association and by the National Science Foundation under grants ATM-9714593 and ATM-0233230. T. Grydeland has been supported through grants 120150/431 and 147769/431 from the NFR of Norway.

EISCAT is an International Association supported by Finland (SA), France (CNRS), the Federal Republic of Germany (MPG), Japan (NIPR), Norway (NFR), Sweden (VR), and the United Kongdom (PPARC).

Topical Editor M. Lester thanks M. Sulzer and another referee for their help in evaluating this paper.

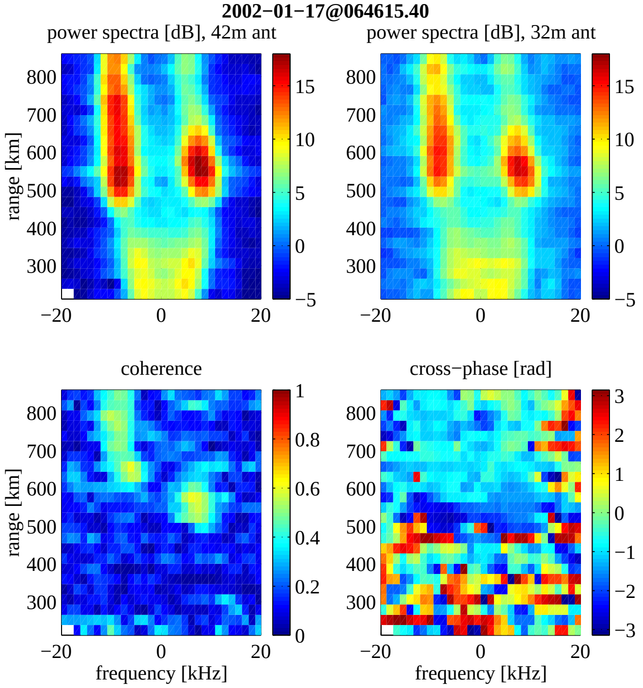

Fig. 13. Incoherent scatter measurements from the EISCAT Svalbard radar using both the field-aligned dish and steerable dish can be used interferometrically to constrain the scale sizes of some types of plasma processes. In this example, naturally occurring ion-acoustic line enhancements are constrained to scale sizes of less than 500 m in the horizontal dimension at the observed ranges.

References

Alker, H.-J.: A programmable correlator module for the EISCAT radar system, Tech. Rep. 79/11, EISCAT Scientific Association, Kiruna, Sweden, 1979.

Bowles, K. L.: Observations of vertical-incidence scatter from the ionosphere at 41 Mc/s, Phys. Rev. Lett., 1, 454–455, doi:10.1103/PhysRevLett.1.454, 1958.

Czechowsky, P., Klostermeyer, J., Röttger, J., Rüster, R., and Schmidt, G.: The Sousy-Svalbard-Radar for middle and lower atmospheric research in the polar region, in Proc. of the 8th Workshop on Techn. Sci. Aspects of MST Radar, edited by Edwards, B., SCOSTEP, Boulder, CO, 1998.

Farley, D. T.: Incoherent scatter correlation function measurements, Radio Sci., 4, 935–953, 1969.

Farley, D. T.: Multiple-pulse incoherent-scatter correlation function measurements, Radio Sci., 7, 661–666, 1972.

Farley, D. T., Ierkic, H. M., and Fejer, B. G.: Radar interferometry: A new technique for studying plasma turbulence in the ionosphere, J. Geophys. Res., 86, 1467–1472, 1981.

Folkestad, K., Hagfors, T., and Westerlund, S.: EISCAT: An updated description of technical characteristics and operational capabilities, Radio Sci., 18, 867–879, 1983.

Gamma, E., Helm, R., Johnson, R., and Vlissides, J.: Design Patterns: Elements of Reusable Object-Oriented Software, Addison-Wesley, ISBN=0-2016-3361-2, 1994.

Grydeland, T., La Hoz, C., Hagfors, T., Blixt, E. M., Saito, S., Strømme, A., and Brekke, A.: Interferometric observations of filamentary structures associated with plasma instability in the auroral ionosphere, Geophys. Res. Lett., 30, 1338, doi:10.1029/2002GL016362, 2003.

Grydeland, T., Blixt, E. M., Løvhaug, U. P., Hagfors, T., La Hoz, C., and Trondsen, T. S.: Interferometric radar observations of filamented structures due to plasma instabilities and their relation to dynamic auroral rays, Ann. Geophys., 22, 1115–1132, 2004, **SRef-ID: 1432-0576/ag/2004-22-1115**.

Hagen, J. B. and Farley, D. T.: Digital correlation techniques in radio science, Radio Sci., 8, 775–784, 1973.

Holt, J. M., Rhoda, D. A., Tetenbaum, D., and van Eyken, A. P.: Optimal analysis of incoherent scatter radar data, Radio Sci., 27, 435–447, 1992.

Holt, J. M., Erickson, P. J., Gorczyca, A. M., and Grydeland, T.: MIDAS-W: a workstation-based incoherent scatter radar data acquisition system, Ann. Geophys., 18, 1231–1241, 2000, **SRef-ID: 1432-0576/ag/2000-18-1231**.

Holt, J. M., Zhang, S.-R., and Buonsanto, M. J.: Regional and local ionospheric models based on Millstone Hill incoherent scatter radar data, Geophys. Res. Lett., 29, 1207, doi:10.1029/2002GL014678, 2002.

Huuskonen, A., Lehtinen, M. S., and Pirttilä, J.: Fractional lags in alternating codes: Improving incoherent scatter measurements by using lag estimates at noninteger multiples of baud length, Radio Sci., 31, 245–261, doi:10.1029/95RS03157, 1996.

Kudeki, E. and Farley, D. T.: Aspect sensitivity of equatorial electrojet irregularities and theoretical implications, J. Geophys. Res., 94, 426–434, 1989.

Lehtinen, M. S. and Häggström, I.: A new modulation principle for incoherent scatter measurements, Radio Sci., 22, 625–634, 1987.

Lehtinen, M. S. and Huuskonen, A.: The use of multipulse zero lag data to improve incoherent scatter radar power profile accuracy, J. Atmos. Terr. Phys., 48, 787–793, doi:10.1016/0021-9169(86)90053-X, 1986.

Lehtinen, M. S. and Huuskonen, A.: General incoherent scatter analysis and GUISDAP, J. Atmos. Terr. Phys., 58, 435–452, doi:10.1016/0021-9169(95)00047-X, 1996.

Lehtinen, M. S., Huuskonen, A., and Pirttilä, J.: First experiences of full-profile analysis with GUISDAP, Ann. Geophys., 14, 1487–1495, 1996, **SRef-ID: 1432-0576/ag/1996-14-1487**.

Lehtinen, M. S., Huuskonen, A., and Markkanen, M.: Randomization of alternating codes: Improving incoherent scatter measurements by reducing correlations of gated autocorrelation function estimates, Radio Sci., 32, 2271–2282, doi:10.1029/97RS02556, 1997.

Mitola, III, J.: Software radio architecture: Object oriented approaches to wireless systems engineering, John Wiley & Sons, ISBN 0-471-38492-5, 2000.

Paul, S., Sabnani, K. K., Lin, J. C.-H., and Bhattacharyya, S.: Reliable multicast transport protocol (RMTP), IEEE J. Selected Areas Comm., 15, 407–421, doi:10.1109/49.564138, 1997.

Reed, J. H.: Software Radio, Communications Engineering and Emerging Technologies Series, Prentice-Hall PTR, ISBN 0-13-081158-0, 2002.

Sahr, J. D. and Lind, F. D.: The Manastash Ridge radar: A passive bistatic radar for upper atmospheric radio science, Radio Sci., 32, 2345–2358, doi:10.1029/97RS02454, 1997.

Saltzer, J. H., Reed, D. P., and Clark, D. D.: End-to-end arguments in system design, ACM Trans. Comp. Sys., 2, 277–288, doi:10.1145/357401.357402, 1984.

Stevens, W. R.: UNIX Network Programming, Prentice-Hall, ISBN 0-134-90012-X, 1998.

Sulzer, M. P.: A radar technique for high range resolution incoherent scatter autocorrelation function measurements utilizing the full average power of klystron radars, Radio Sci., 21, 1033–1040, 1986.

Sulzer, M. P.: A new type of alternating code for incoherent scatter measurements, Radio Sci., 28, 995–1001, 1993.

Turunen, T.: GEN-SYSTEM—a new experimental philosophy for EISCAT radars, J. Atmos. Terr. Phys., 48, 777–785, doi:10.1016/0021-9169(86)90052-8, 1986.

Turunen, T. and Silén, J.: Modulation patterns for the EISCAT incoherent scatter radar, J. Atmos. Terr. Phys., 46, 593–599, doi:10.1016/0021-9169(84)90076-X, 1984.

Turunen, T., Markkanen, J., and van Eyken, A. P.: Ground clutter cancellation in incoherent radars: solutions for EISCAT Svalbard radar, Ann. Geophys., 18, 1242–1247, 2000, **SRef-ID: 1432-0576/ag/2000-18-1242**.

Uppala, S. V. and Sahr, J. D.: Aperiodic transmitter waveforms for spectrum estimation of moderately overspread targets: new codes and a design rule, IEEE Trans. Geosci. Remote Sensing, 34, 1285–1287, doi:10.1109/36.536545, 1996.

Wannberg, G.: The G2-System and general purpose alternating code experiments for EISCAT, J. Atmos. Terr. Phys., 55, 543–557, doi:10.1016/0021-9169(93)90003-H, 1993.

Wannberg, G., Wolf, I., Vanhainen, L.-G., Koskenniemi, K., Röttger, J., Postila, M., Markkanen, J., Jacobsen, R., Stenberg, A., Larsen, R., Eliassen, S., Heck, S., and Huuskonen, A.: The EISCAT Svalbard radar: a case study in modern incoherent scatter radar system design, Radio Sci., 32, 2283–2307, doi:10.1029/97RS01803, 1997.