I, John David Jacobs, hereby submit this as part of the requirements for the degree of:

Master of Science

in:

Systems Engineering, E.E.

It is entitled:

Hyperspectral Planner Instrumentation for Product Goal Synthesis in Material Process Control

Approved by:

Dr. Patrick H. Garrett

Dr. William G. Wee

Dr. David B. Mast
HYPER SPECTRAL PLANNER INSTRUMENTATION FOR PRODUCT GOAL SYNTHESIS IN MATERIAL PROCESS CONTROL

A thesis submitted to the Division of Research and Advanced Studies of the University of Cincinnati in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in the Department of Electrical and Computer Engineering and Computer Science of the College of Engineering

Two Thousand One

by

John David Jacobs

B.S.E.E., University of Cincinnati, 1999

Committee Chair: Patrick H. Garrett
Abstract

Motivated by a tri-level hierarchical process control scheme developed at the Wright Patterson Air Force Base materials research laboratory, this thesis explores contemporary instrumentation methods to accommodate each level of the process control structure. The three hierarchical process levels include an environmental subprocess level, an in situ subprocess level, and an ex situ subprocess level which are linked by physical influences defining the transport of energy, mass, and momentum.

A signal model hierarchy is introduced to delineate appropriate signal attributes useful for developing economical instrumentation at the corresponding control level. This model clearly demonstrates an increase in signal complexity and computational requirement with ascension from environmental to ex situ processing levels, which translates into an equivalent computational capacity required at each level of instrumentation implementation.

An instrumentation taxonomy is characterized based on the requirements observed from the signal model. This structured instrumentation taxonomy defines contemporary instrumentation by application, method, and performance requirement to serve as a guide for making prudent and economical design decisions. This classification is validated by detailed design contribution examples each illustrating a
level in the overall taxonomy. The resulting description lays a foundation for the design of high-performance instrumentation.

This thesis concludes with a description of high-performance instrumentation for ex situ planners as applied to material process control applications. Such devices, termed analytical instruments, represent the upper-echelon of high-performance, computationally capable instrumentation as described herein. This level of instrumentation is able to realize the appropriate algorithms for product microstructural interpretation and goal-product comparisons, at the necessary bandwidth. With the use of this advanced instrumentation, it is possible for ex situ planners to be designed for optimal real-time product evaluation thus enabling automatic redirection of drifting process parameters.

The term hyperspectral imaging is used to describe the versatile collateral method for integrating both spatially and spectrally continuous sensor data simultaneously to assist product characterization. The ex situ planner instrumentation utilizes this multisensor data to compute a product facsimile and update controller references based on product goal comparisons.
Preface

I want to thank my advisor Dr. Patrick Garrett for his continuous guidance and encouragement throughout this thesis. Special thanks go to Dr. David Mast of the University of Cincinnati Physics Department for his contributions and counseling with the microwave microscopy and AC magnetization projects. I also thank Dr. William Wee for participating on my thesis committee, along with Dr. Patrick Garrett and Dr. David Mast.

This is dedicated to my family and friends whose endless support and encouragement have empowered me in all of my efforts.
Contents

Contents .............................................................................................................................. 1
List of Tables and Figures .................................................................................................. 2
Chapter 1: Statement of Problem ................................................................................... 3
   1-1 Background: An Overview of the Hierarchical Process Control Schema .......... 5
Chapter 2: Method of Solution ....................................................................................... 10
   2-1 Instrumentation Systems Taxonomy .................................................................. 15
Chapter 3: Discrete Instruments ................................................................................... 21
   3-1 System for Extracting the AC Magnetization Characteristics, Susceptance and Loss, In Superconductive Materials ......................................................... 23
   3-2 Data Gathering Procedure ............................................................................... 23
   3-3 The Discrete Instrumentation System ............................................................... 24
Chapter 4: Virtual Instruments ..................................................................................... 27
   4-1 Microwave Microscopy Virtual Instrument ..................................................... 29
   4-2 Graphical Programming Software .................................................................... 39
   4-3 PCI Bus ............................................................................................................. 46
Chapter 5: Analytical Instrumentation In Advanced Control ..................................... 48
   5-1 Autoassociative Crystal Feature Extraction .................................................. 50
   5-2 DSP Hyperspectral Integration of Multisensor Data ........................................ 55
Conclusions ...................................................................................................................... 57
Bibliography .................................................................................................................... 59
Appendix I ....................................................................................................................... 61
   The AC Magnetization Software Control Algorithm ........................................... 61
Appendix II ..................................................................................................................... 70
   The Microwave Microscopy Software Data Acquisition Algorithm .................... 70
Appendix III ..................................................................................................................... 73
   Introductory Tutorial of Autoassociative Neural Networks .................................. 73
List of Tables and Figures

Table 4-1. Graphical Programming Software Packages ................................. 28
Table 4-2. Signal Glossary ............................................................................ 33

Figure 1-1. Hierarchical Subprocess Control Schema ................................. 8
Figure 2-1. Multisensor Signal Model Hierarchy ........................................ 12
Figure 2-2. Hierarchical Instrumentation Taxonomy .................................... 16
Figure 2-3. Computer Bus Classification ..................................................... 17
Figure 3-1. IEEE-488 GPIB ................................................................. 22
Figure 3-2. Superconductive Magnetization Material Characterization Instr. .... 26
Figure 4-1. Microwave Microscopy System Diagram .................................. 31
Figure 4-2. Microwave Microscopy Virtual Instrument .............................. 37
Figure 4-3. Pseudo-Measure of Q and Resonant Frequency Shift Images ....... 38
Figure 4-4. Graphical User Interface For Displaying Microwave Data ........... 41
Figure 4-5. LabVIEW Display Generation Graphical Program .................... 44
Figure 4-6. Microwave Microscope Algorithm Module Hierarchy ............... 45
Figure 4-7. Exemplar PCI Bus Structure .................................................... 47
Figure 5-1. Hyperspectral Analytical Instrumentation System ..................... 49
Figure 5-2. Autoassociative Feature Crystal Classification ......................... 51
Figure 5-3. Single Model of Neuron .......................................................... 52
Figure 5-4. Pyramidal Structure of the Autoassociative Neural Network ....... 53
Chapter 1: Statement of Problem

Technology increments have accommodated new capabilities in the area of instrumentation and control. Embedded computers, high-speed bus structures, and digital signal processing (DSP) techniques contribute to the emerging flux in computational capacity. Improved silicon processes have provided industry with powerful processors at higher operation speeds and gate counts. Recent micro-miniaturization techniques (i.e. MEMS) have expanded the production of low-cost, small form-factor, sensors and actuators. Prevalent advancements have broken limiting barriers by allowing computationally complex control and instrumentation methods to be realized and higher performance specifications to be met, which have especially been of benefit to complex disciplines including the processing of advanced engineering materials.

Advanced material process control has traditionally exploited leading instrumentation technology to produce materials that require processing in environments far from equilibrium and meet difficult specifications. The need for advanced measurement instruments elicits the need for integration, and a control structure to orchestrate process parameter trajectories to achieve the specifications of the goal material. Such a controller is not trivial. Over many years of research a hierarchical subprocess control schema has evolved which has proven beneficial over a wide range of complicated process applications. This controller consists of three distinct levels of
control: environmental, in situ, and ex situ levels. Each ascending level raises the attribution of the overall controller to actively compensate and effectively attenuate variability, disorder and disturbance dynamics that occur throughout a process cycle.

In practice, the ex situ-level controller is usually difficult to realize. Computational complexities have historically limited ex situ control to post-processing reference corrections. These offline compensations, based on complicated and frequently incomplete mathematical models, detract from the full potential a real-time hierarchical controller could achieve. With contemporary advancements in technology, ex situ planners can be designed for optimized real-time product evaluation and automatic redirection of drifting process parameters.

This thesis develops advanced analytical instrumentation architecture as required in advanced materials synthesis. In its development, a classification structure for modern instrumentation methodologies is presented through a structured instrumentation taxonomy that delineates contemporary instrumentation by application, method, and performance requirement. This classification is validated by detailed design contribution examples each illustrating a level in the overall taxonomy. The resulting description lays a foundation for the design of high-performance instrumentation and also provides a unique perspective.
1-1 Background: An Overview of the Hierarchical Process Control Schema

Challenges to contemporary process control include realizing the potential of in situ sensors and actuators applied beyond apparatus boundaries to accommodate increasingly complex process operations. The relationship between process and control design generally involves process design for controllability with stability provided by the control compensator design. Accurate process modeling and control compensation includes effective attenuation of variability, disorder, and disturbance dynamics which is aided by process decomposition into a natural hierarchy of linear and decoupled influences that link environmental, in situ and product subprocesses. Real-time process measurements offer both model updating and minimization of processing disorder through feedback regulation. Further, accurate process models enable optimum feedforward control references for achieving reduced-disturbance state progression throughout a processing cycle.

Process models provide useful describing functions of state transitions and knowledge of how process influences are affected throughout a processing cycle. Benefit accrues from process decomposition into subprocesses to enable more precise modeling regions. The degree of decomposition depends upon the achievable modeling granularity and the natural process hierarchy encountered. Of primary interest is the capture of subprocess dynamics that predict the progression of states for control purposes.
The limitations of traditional methods of material process control include modeling that is essentially confined to the description of energy, mass and momentum transfers at process apparatus boundaries, and inefficient control compensations arising from the need to iteratively regulate and relinearize process parameters, while simultaneously meeting linear control stability and response requisites.

The hierarchical process control schema of Figure 1-1 shows a system structured according to an increasing process knowledge representation at decreasing accuracy with subprocess ascension, analogous to Heisenberg’s uncertainty principle. This tri-level process control scheme is delineated by environmental-, in situ-, and ex situ-levels. The environmental controller employs first-principles to control mass-flow, temperatures, and pressures at the apparatus boundaries of the product subprocesses. Measurement and actuation extending beyond process apparatus boundaries with control compensation of subprocess influences, constitutes the in situ level. The employed in situ sensors are chosen to extract electrical, chemical, and thermal feature data evolving during the material process cycle. The in situ compensator uses this multisensor data as well as references generated by the ex situ planner to compute environmental set points and affect the process state trajectories in effort to achieve optimal material products. The ex situ planner incorporates multiple knowledge sources to emulate a feedforward control structure which in turn computes the in situ references and manages remodeling/adaptation in response to process progression. It asserts itself as the most computationally intelligent level in the process control.
hierarchy and employs analytical models, heuristic rules, and empirical data knowledge sources.

The ex situ planner plays the role of supervisor as it orchestrates the material process cycle by predicting and manipulating in situ references. It provides a function approximating the inverse of the aggregated subprocess models of Figure 1-1, where planner outputs coincide with in situ references, and planner inputs include environmental, in situ, and material subprocess data. The merit of ex situ feedforward control is the prediction of in situ reference values by a modeling function that can compensate for slowly varying process disturbances of in situ state-variables relative to their goal paths. This contrasts with feedback control, which is suitable for regulating rapidly varying disturbances.

Elements of the ex situ planner include a comprehensive process description function, capable of defining aggregate process behavior throughout a processing cycle with greater robustness but less precision, for example, than in situ sub process models. A frequent limitation is the inability of mathematical process models to represent process element interrelationships. The planner may therefore be augmented by process measurement data and subsequent product microstructural interpretation. The latter relates material properties to corresponding process influences, which may lend itself to a particular qualitative or symbolic method.
Figure 1-1. Hierarchical Process Control Schema
To summarize, the three hierarchical process levels include an environmental subprocess level, in situ subprocess level, and ex situ subprocess level which are linked by physical influences defining the transport of energy, mass, and momentum. Influence descriptions beneficially extend process representations to more definitive subprocess cause and effect attributes beyond traditional apparatus boundaries.

Primary issues associated with this hierarchical subprocess structure include:

1. Identification at each level with sufficiently accurate variable modeling for state progression throughout a processing cycle to enable the implementation of feedback or feedforward control compensators with adequate performance for complex process applications.

2. A companion discovery capability for detecting and interpreting state excitation noise that influences process variable disturbances with mechanisms for their counteraction.
Chapter 2: Method of Solution

Figure 2-1 shows the organization of models and corresponding signal sets, which constitute the hierarchical process control system introduced in Figure 1-1. The three rows represent the environmental-, in situ-, and ex situ-levels (ascending from bottom to top). The right-most column illustrates the signal flow between the stages of the controller. The left-most elements describe the three signal-modeling techniques used for description of environmental signals, in situ states, and real-time product features. The central column shows signals that exemplify its respective level.

Environmental signals are quantities that can be measured and actuated at the process boundaries such as temperatures, pressures, flow rates, etc. In situ signals are quantities that are measured beyond the process boundaries. In essence, they are those quantities that are naturally a part of the process dynamics, and occur “within” the process boundaries. They include evolving process parameters like chemical composition, morphological properties, and electrical and mechanical attributes. They characterize the measurable state dynamics of the process. Ex situ signals include comprehensive multisensor data computed from the in situ and analytical sensor signals. Analytical sensors augment the in situ signals by detecting product features outside the confines of the processing apparatus. The difference between the in situ and the analytical sensors of the ex situ subprocess has traditionally been a matter of location. In situ sensors measure events occurring inside the process. Analytical
sensors measure product attributes after the product cycle has completed. It is notable that real-time process control only employ in situ sensor signals at the ex situ subprocess level. Ex situ signals are multi-source sensor data utilized to compute and envisage evolving material features that cannot directly be measured via a single sensor, hence the name ex situ-level since the data is derived through vehicles outside the physical process. This data is of higher cognitive content and can be used in conjunction with experimental databases to interpret and predict product outcome and affect corrective actions to achieve congruence with desired product goals. It is notable that the inadequacy of some analytical process models has placed an emphasis on sensors in a substitutive role to provide comprehensive process characterization.
Figure 2-1. Multisensor Signal Model Hierarchy
The usefulness of the signal model hierarchy is evident when considering the instrumentation choices involved in implementation. Sensor and actuator complexity increases with ascension of the three levels of the signal model. This diversity of signal complexity elicits a subsequent need for a categorical description of instrumentation methods tailored to the bandwidth and computational needs of the application of interest. This multisensor process control signal model hierarchy reflects the general signal attributes common to a wide range of process control applications. Furthermore, it permits a corresponding instrumentation taxonomy to be formulated whose purpose is to serve as a guide for making optimal instrumentation implementation decisions.

Environmental signals can be modeled with single-input-single-output (SISO) linear systems described by constant coefficient ordinary differential equations, which neatly lend to an input/output transfer function representation. Many environmental signals can be sufficiently modeled with first-order Laplace transforms and a time delay approximation. In situ signals represent the measurable knowledge content inside the process as it evolves. When possible the in situ signal model analytically captures process dynamics, such as mass transfer chemical reactions and material microstructure, in terms of the composite of measurable in situ and environmental signals. In situ signal data is typically multidimensional; therefore multi-input-multi-output (MIMO) state space models are effectively used. Examples of in situ sensors include the mass spectrometer, microbalance thickness sensor, and Raman spectrometer. The ex situ subprocess level processes in situ and analytical sensor data
to predict and/or analyze product features. Examples of analytical sensors include the scanning electron microscope (SEM) and evanescent microwave microscope (could be used as either an in situ or an ex situ sensor). The ex situ feature model uses symbolic methods such as rule-based heuristics, an artificial neural network, or other approach to extract attributes and interpretations of the developing product. This signal data composition at the ex situ level is of high cognitive value. The computed feature set often represents a virtual product facsimile and/or describes the relevant specifications that correspond to the predicted quality of the product outcome. This data is beneficially used to guide the process towards optimal goal states.

It is apparent that the analytical and in situ sensors as well as the processing instrumentation at the ex situ level will require more computational power than the environmental signal (transducer-based) counterpart. This indicates an obvious delineation in computational requirement and therefore the three constituent subprocess implementations should reflect this in order to keep the instrumentation economical and to avoid “over-“ or “under-“ designing mishaps. The following section introduces an instrumentation systems taxonomy to serve as a design guide for choosing effective implementation strategies wisely.
2-1 Instrumentation Systems Taxonomy

The evolution of electrical measurements spans two centuries marked by the invention of
the galvanometer in 1820. Continued development has provided an expanding range of
sophisticated measurement, signal conditioning, analysis, and data presentation
capabilities with an instrumentation taxonomy, represented by Figure 2-2, that can
accommodate the comprehensive data requirements of advanced hierarchical sensor and
actuator systems. Three distinct instrumentation architectures are defined by the structure
shown, each of which involve different implementations for meeting their respective
stimulus and measurement applications. Examples are presented in the sections that follow
for each of these architectures that highlight effective approaches to contemporary
instrumentation challenges.
Figure 2-2. Hierarchical Instrumentation Taxonomy
The diversity of existing bus structures provides a useful delineation of capabilities for instrumentation system integration. Figure 2-3 introduces basic computer bus classifications. Level-0 traces describe inter-component board connections that are characterized by signals specific to their digital devices. Level-1 dedicated buses, such as the industry standard architecture (ISA) bus, provide buffered subsystem peripheral component interfacing including protocols to accommodate signal propagation delays. Level-2 system buses, such as the peripheral component interconnect (PCI) structure offer comprehensive bus master services including

Figure 2-3. Computer Bus Classification
arbitration and concurrent operation. Level-3 parallel buses enable peripheral extensions for Level-1 buses including the general-purpose interface bus (GPIB) and small computer systems interface (SCSI) bus. Level-4 serial buses are the longest structures in the bus repertoire, and range from early standards such as RS-232C to the more recent universal serial bus (USB) and IEEE-1394 (FireWire). Serial bus transmission protocols are divided into isochronous (real-time), synchronous, and asynchronous modes with the latter prevalent. The Level-5 video bus may be limited to an accelerated graphics port (AGP), which supports video display devices.

Gigabit Ethernet can aid external information exchanges with the host computer for all of the instrumentation architectures of Figure 2-2, especially when high-resolution graphics are involved. The efficiency of Gigabit Ethernet relies upon full-duplex transmission employing all four-wire pairs of common Category 5 cable, plus enabling terminal equipment shown in Figure 9-4. Performance is facilitated by 5-level PAM coding, Trellis forward error correction, and DSP received signal equalization. Conventional Ethernet parameters are also introduced in the following section.
The instrumentation taxonomy levels can be described as follows:

**Discrete Instruments** are those devices that are designed to perform specific analytical functions and communicate over a standardized industrial bus structure (i.e. IEEE-488, HPIB, etc.). In general, they are easy to use yet lack the versatility and bandwidth for high-speed customizable applications. Examples of discrete instruments include oscilloscopes, digital multimeters, spectrum analyzers, and temperature controllers.

**Virtual Instruments** are devices, such as a data acquisition (DAQ) board, that plug into a host computer via a bus interface. The term "virtual instrument" is coined from the versatility the I/O boards provide when combined with the host software. For example, a typical DAQ board can be configured to perform analysis and capture functions, control motors, generate waveforms, etc. Its functionality is reconfigurable and dependent on the software that controls it. The control structures and algorithms, which are implemented using graphical/scripted code on the host computer, are limited by the bandwidth of the I/O board, the speed of the host computer hardware, and most importantly the overhead of the operating system running the control software. Therefore real-time control architectures that require mathematically intensive computations or deterministic timing cycles run into a performance barrier due to these factors. It becomes difficult to predetermine performance bottlenecks when constructing a control system with multiple I/O boards and various types of computations. Virtual instruments are best used when timing requirements are non-
critical or the required bandwidth is deemed sufficiently low so that the host computations can be implemented and performance criteria are met. With the advent of higher performance computer processors and bus architectures, these limitations will begin to vanish.

**Analytical Instruments** are a category representing the highest-performance instruments where computationally intensive calculations must be made in guaranteed time intervals and/or at relatively high speeds. Signal processing, adaptive systems, computationally intelligent controllers, etc. are general application classes, which exemplify such instruments. These devices are programmed and operate such that all the computationally intensive functions are completed locally via specialized hardware (i.e. DSP, ASICs, μC, etc.). The computation/mathematical capacity is "integrated" and managed by the local hardware. This guarantees robust, reliable, real-time control during every operation. The host computer is freed of the computational burden and only orchestrates the integration of the components of the overall control architecture to provide user interface operations, networking, and data storage.
Chapter 3: Discrete Instruments

Discrete instruments are devices designed to perform specific analytical functions. The user has minimal influence to reconfigure the device’s function other than what was intended by the manufacturer. Functions performed by such equipment are not user-application specific. Examples of discrete instruments include oscilloscopes, spectrum analyzers, and digital multimeters.

All useful instruments provide an interface mechanism for sending and receiving data in a control system. Discrete instruments typically have industry standard ports for data exchange. The most common is the IEEE-488 general-purpose interface bus (GPIB).

The GPIB bus has achieved acceptance since its introduction by Hewlett-Packard because of its robustness for networking discrete instruments. This parallel bus can link 15 instruments plus a controller with 16 active lines, 8 for data and 8 for control, shown in Figure 3-1. Communication control procedures initiated prior to data transmission designates transmitting instruments and receiving instruments. Instead of address lines there are three data-transfer and five bus-management lines for communication utilities. When ATN is high all instruments must listen to the DIO lines. When ATN is low only designated instruments can send and receive data. The maximum throughput data rate is around 1Mbyte/s.
Figure 3-1. IEEE-488 GPIB.
3-1 System for Extracting the AC Magnetization Characteristics, Susceptance and Loss, In Superconductive Materials

Essential to the development and study of new materials is the ability to characterize their electrical, magnetic, and mechanical properties. This example describes an instrumentation system that measures AC magnetic susceptance and loss for high-temperature superconductive materials as a function of an applied magnetic field and temperature. Susceptibility is the measure of how the internal magnetic fields of a material affect the permeability of the material to an externally applied magnetic field. Internal magnetic fields arise from electrons orbiting around nuclei or by electrons spinning around themselves. Loss is the term used to describe the energy dissipation in the material due to the oscillating magnetic field. Physicists use this data as a figure of merit for determining the quality of superconductive samples.

3-2 Data Gathering Procedure

The data gathering procedure involves first cooling a material sample to temperatures where the sample becomes superconducting. The sample is placed in an apparatus that generates an external magnetic field surrounding the sample. This field is controlled so that it oscillates at a fixed frequency but at user-defined amplitudes. A heater element and a temperature transducer are also present so that the sample’s
temperature can be controlled and monitored. Heat is applied so that the sample
temperature will drift upwards. As the temperature rises, the susceptibility meter
measures both the susceptance and loss for various amplitudes of the applied
magnetic field over the desired temperature range. From this process, a series of data
images (i.e. each curve corresponding to unique magnetic field amplitudes) are
plotted to characterize the magnetic properties of a superconductive sample as a
function of temperature.

3-3 The Discrete Instrumentation System

The following instruments were used in the process and software implementation:

- Lakeshore Temperature Controller – IEEE-488 Interface
- Keithley 197 Digital Multimeter (DMM) – IEEE-488 Interface
- Keithley 175 DMM– IEEE-488 Interface
- AC Susceptance Meter – Analog Voltage Interface
- National Instruments Analog Output (AO) Board

The first four devices are classified as discrete instrumentation. The analog output
device is considered a virtual instrument (see the chapter on virtual instruments).
National Instruments LabVIEW 5.1 was the software package used to implement the controller. It allowed for easy interface of all the above devices. The resulting application was designed for use on a PC platform running Windows NT.

The principle excitation in this process is the heat delivered by the temperature controller once the test sample is cooled to the desired initial temperature. The heater power of the temperature controller must be controlled so that the thermal drift rate is sufficient for data to be accurately measured and recorded for each of the magnetic field amplitudes.

Figure 3-2 describes the system connectivity for the AC magnetization instrumentation.¹

¹ Appendix I contains more information on the control software and its features.
Figure 3-2. Superconductive Material Magnetization Characterization Instrumentation
Chapter 4: Virtual Instruments

The concept of computer-based instruments arose at the beginning of the 1970s with the advent of inexpensive computation, furthered by the personal computer that permitted networking discrete instruments into sophisticated automated test systems. The evolution of more efficient data acquisition and presentation, resulting from user defined programmability and reconfigurability, continues through the present to provide a more computationally intensive instrumentation framework. Contemporary virtual instruments accordingly are capable of elevating fundamental sensor data to a substantially higher attribution enabling more complex cognitive interpretation. Multifunction I/O hardware is typically combined with application development software on a personal computer platform for the realization of specific virtual instruments like the microwave microscopy example for sample assays in manufacturing and biomedical applications.

The architecture of virtual instrument software may be divided into two layers: measurement and configuration services, and application development tools. Measurement and configuration services contain prescriptive software drivers for interfacing hardware I/O devices as subroutines that are usually accessed by graphical icons. Configuration utilities are also included in this layer for naming and setting hardware channel attributes such as amplitude scaling. Software selected for application development may be sourced separately from hardware devices only when
compatibility is insured. Examples of commercial virtual instrument software are
listed in Table 4-1 for data acquisition, processing, presentation, and communications
tasks. Graphical languages have become dominant for these systems owing to their
speed of system prototyping, ease of data presentation, and self-documentation.

<table>
<thead>
<tr>
<th>Vendor</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Labtech</td>
<td>Notebook Icon-based data acquisition and analysis software.</td>
</tr>
<tr>
<td>DaDISP</td>
<td>Waveform analysis and display software.</td>
</tr>
<tr>
<td>LabVIEW</td>
<td>Software suite that provides a graphical programming environment for data acquisition applications.</td>
</tr>
<tr>
<td>Agilant VEE</td>
<td>Graphical programming environment that incorporates MATLAB for versatile data acquisition and analysis.</td>
</tr>
<tr>
<td>SNAP MASTER</td>
<td>Data acquisition, analysis, and control software.</td>
</tr>
<tr>
<td>SoftWIRE</td>
<td>Visual Basic add-on software to facilitate graphically based creation of application software.</td>
</tr>
<tr>
<td>Visual Basic, C++</td>
<td>These languages are often incorporated into the above data acquisition systems in various ways.</td>
</tr>
</tbody>
</table>
Microwave microscopy is a novel approach to nondestructive/non-contact imaging with a very high-resolution capacity. Such a device is useful for studying material variations and characteristics at submicron levels without damaging or touching the material sample. One of the main advantages of microwave microscopy is its larger penetration depth, which makes a sample’s subsurface properties detectable as well as the surface properties.

To achieve submicron resolutions using microwaves, evanescent electromagnetic fields must be used to surpass the half-wavelength limit of minimum resolvable size known as the Abbe barrier (on the order of centimeters for microwaves). In this so-called near field, evanescent microwaves decay exponentially in space. As a result, they contain frequencies higher than $c/\lambda$, are more confined than single tone sinusoidal wave and therefore get around the Abbe limit of resolution for near field probes (instrument sensitivity and probe geometry are the practical limits of the minimum resolvable size of a material sample).

A microwave microscopy probe (MMP) operates by employing a microwave resonator structure (microstripline, coaxial, or other geometrical configuration) as a sensor. When an object is placed at the tip of the resonator, the reflection coefficient of the probe changes. Both the resonant frequency and the quality factor of the
resonator are a function of the material sample and its distance from the tip of the probe. Various signal detection methods can be employed to detect the microwave properties of the material under the MMP. Typically one can modulate the excitation frequency and monitor the change in the reflection amplitude. The excitation frequency is chosen to yield the maximum change in the probe’s reflection coefficient at a resonant mode of microwave probe.

The microwave microscopy virtual instrument system consists of the following components:

1. Host PC Computer
2. National Instruments Analog/Digital I/O Board
3. RS-232 Serial Port
4. Stepper Motors with an XY Positional Platform
5. $\frac{1}{2}\lambda$ Coaxial Microwave Resonator (Sensor)
6. Ferrite Microwave Circulator Coupling
7. Microwave Signal Generator
8. Crystal Microwave Detector
9. Microwave Power Amplifier
10. Reflected Amplitude Signal Filter (3-pole band-pass filter)
Figure 4-1. Microwave Microscopy System Diagram
The host computer provides all of the control signals for the XY platform stepper motors, generates the modulation signal for excitation of the MMP, and acquires the reflected amplitude signal. One byte of I/O signals in conjunction with one RS-232 serial interface is used to control two stepper motors that position the XY platform beneath the sensor. The microwave generator produces a single tone excitation proportional to an input DC voltage. Applying a sinusoid signal to the control input of the microwave generator will cause it to sweep frequencies proportional to the control input’s peak-to-peak amplitude. This allows for efficient control and frequency modulation of excitation signals. The crystal microwave detector operates by rectifying the sinusoidal reflected signal, which produces a DC voltage proportional to the amplitude of the reflected wave. The host computer fulfills the signal processing requirements via a National Instruments DAQ (Data Acquisition) I/O board that plugs into an available PCI bus slot. This DAQ board accommodates all of the analog and digital I/O requirements.

Various external hardware devices are also used in this system. A ferrite “circulator” coupling is used to couple microwave energy to the $\frac{1}{2}\lambda$ coaxial microwave resonator. A circulator is a 3-port device that allows one to input an excitation signal, apply this signal to a load, and obtain the reflected signal. The excitation from the output port of the circulator is weakly coupled to the microwave resonator using a coupling that provides an adjustable capacitance.
A microwave amplifier is used to boost the excitation signal before its applied to the microwave resonator. This is done to obtain an appropriate reflected signal SNR since at resonant frequencies the attenuation of this signal is large (-40dB). Before the host computer acquires the reflected amplitude signal, it is band-pass filtered using a 3-pole analog filter to prevent aliasing and to suppress high frequency noise.

A software algorithm is used to control and acquire microwave data for a given microscopy sample. Two types of information are collected and plotted in three axes. One is a measure of the resonant frequency displacement from the unloaded resonant frequency ($f_0$). The other is a measure of the quality factor or selectivity of the sensor loaded by its respective position over a sample.

<table>
<thead>
<tr>
<th>TABLE 4-2. Signal Glossary</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Excitation Signal</strong></td>
</tr>
<tr>
<td><strong>1F Signal</strong></td>
</tr>
<tr>
<td><strong>2F Signal</strong></td>
</tr>
</tbody>
</table>
As alluded to above, the 1F signal frequency modulates the excitation signal so that a frequency range centered about the resonant frequency of the unloaded MMP is swept. By sweeping these frequencies a characteristic measure of the reflection coefficient can be obtained. If the offset of the 1F signal is chosen such that the swept frequencies are centered about the resonant frequency then the 2F signal will be periodic with only 2f-frequency components. One cycle of the 1F signal corresponds to two peaks (the left and right most of the resonant curve) and two minima (crosses the minima of the resonant curve twice) along the resonant characteristic. Hence the names of the signals, 1F and 2F.

When the excitation signal is not symmetrically modulated about the resonant frequency of the MMP, the 2F signal will contain both f and 2f frequency components. Through signal processing of the acquired 2F signal, the amplitude of the 2f component is determined (denoted |2F|2f). The 1F signal is also sampled, and the amplitude of the f component is computed (denoted |1F|). These amplitudes are then used in a proportional-integral (PI) feedback system to affect the DC offset of the 1F signal (i.e. shift the center frequency of the excitation). The PI loop ceases when its error signal is within a predetermined dead band region for a preset amount of time. When this happens the 2f-frequency component of the 2F signal will be maximized and the f frequency component will be minimized. This implies that the center frequency has converged to a new resonant frequency value given the interaction between the MMP and the material beneath it. At this time the DC value, which corresponds to the shift in resonant frequency from the initial value, is
recorded. The amplitude of the $2f$ is also recorded. It provides the pseudo measure of Q at that sample point. More specifically, it is a metric for the steepness, or selectivity, of the new resonant curve over the fixed band of frequencies swept by the 1F signal amplitude. The greater the 2F signal amplitude, the more selective the resonator has become at its new resonant frequency due to its location over the scanned material.

Instead of Q, this quantity is often referred to as loss. When electromagnetic energy is “lost” due to absorption by the sample material, less energy will be reflected. Hence the reflection coefficient will be smaller at the resonant frequency and therefore more selective. Since the term “loss” and “Q” both refer to the selectivity of the resonator, either term can be used.

Figure 4-2 collectively describes the software signal processing operations performed for each point over the material specimen. Sampled data vectors of the 1F and 2F signals are captured. Both data vectors are then high-pass filtered in software to remove any offset components. Described by equation 4-1, the discrete time Fourier series (DTFS) magnitude at frequency $f$ is estimated for the 1F signal and at frequency $2f$ for the 2F signal. These values then comprise the error signal in the PI control loop which periodically adjusts the DC offset of the 1F signal thereby seeking the new resonant frequency associated with the current test material coordinates. The PI ceases adjustment of the DC offset when the error signal shrinks within a
predefined dead band region. The iterative data capture and signal processing then terminates, the software then compiles and updates the data display, and finally the MMP moves to the next location over the material sample. The process repeats until the material has been scanned entirely.

\[
M(f_m, x) = \frac{2}{N} \sqrt{\sum_{i=1}^{N} x_i \cdot \sin \left( \frac{2\pi \cdot i \cdot f_m}{f_i} \right)^2} + \sqrt{\sum_{i=1}^{N} x_i \cdot \cos \left( \frac{2\pi \cdot i \cdot f_m}{f_s} \right)^2}
\]

Equation 4-1
Figure 4-2. Microwave Microscopy Virtual Instrument
Below is an example of the graphical data generated from a 2mm square surface scanning of black titanium at 25µm per step. The source frequency was 9.670GHz. The 1F amplitude (modulation amplitude) was 8Volts peak to peak at a 100Hz. The sampling rate was 10000 samples per second. The modulation sensitivity was 6V/MHz. Both the measure of Q and the resonant frequency shift profiles are shown as a function of position over the material sample.

This application’s main task was to sample/generate signals having harmonics less than 5KHz. The mathematical complexities involved with the algorithms performed were limited to averaging and comparison functions. None of which were time-critical or mathematically intensive enough to require a real-time hardware implementation, since the algorithm operates point-by-point over the material under test. Therefore, the choice of implementing this application using virtual instrumentation was an economical one.
4-2 Graphical Programming Software

The microwave microscopy virtual instrument software was realized using National Instruments LabVIEW version 5.1. LabVIEW is a popular program development package designed specifically for instrumentation and data acquisition applications. LabVIEW is different from other methods of programming, such as C/C++ or BASIC. These programming systems traditionally use text-based languages to create lines of code, whereas LabVIEW uses a graphical programming language to create programs. Programming using graphical modules elicits numerous advantages. For non-programmers, the creation of a control program becomes more visual and thereby providing a more intuitive software design process. Since LabVIEW contains an extensive library of functions and subroutines it shortens the design time. Instead of writing numerous lines of code, the code is already embedded into the graphical modules. Once the algorithm is designed, only the time needed to hook the graphical elements together is required. Since the design environment is graphical, the program structure naturally takes on a hierarchical structure, which makes for easier debugging. Furthermore, the modular nature of the programs allows for the abstraction of complicated algorithms into easier sub-modules. These modules can then be completed individually and wired together to form the final algorithm.
In addition, LabVIEW also contains application-specific libraries for data acquisition, GPIB and serial instrument control, data analysis, data presentation, and data storage. Many of these special functions were employed in this software.

Each LabVIEW program consists of two entities, a diagram and a front panel. The diagram acts as the source code for the program. The front panel is a graphical user interface that allows data to be displayed and control variables to be affected. Once the program is initiated, all graphical instructions are executed concurrently. The only exception is, data elements that feed information into other data structures are executed first in succession.
The figure above depicts the software module that displays the scanned microwave data to the user. In this example, the 3D graph is presenting the resonant frequency shift information for a 2mm square sample of blue titanium. The subsequent figure shows the associated graphical program that executes to provide the user interface and 3D graphic. This program allows the user to print the graphic data, export the data as a Matlab file, load previously saved data, and view the 3D data as a 2D intensity graph.

Figure 4–4. The Graphical User Interface (GUI) For Displaying 3D-Microwave Information
Figure 4-5 is an example of graphical code. When the program is initiated, the group of icons labeled (1) is set true. These icons define the display visibility attributes for the buttons on the front panel. Then the outer-most window is initiated. This window is a “while loop”. Everything inside is executed repetitively until control “Done” is set false, enumerated (2), allowing conditional expressions to break this ‘while loop’. The metronome icon is a millisecond timer that controls the speed interval at which “while loops” iterate. Here it is set to loop no faster than every 50 milliseconds. The next concentric window is analogous to a “case” statement in the C programming language. The Boolean variable labeled “Load” controls the execution of this “case” window.

The group of icons labeled (3) allows the user to locate the path of an image file. The results of the previously generated file path control yet another “case” window. Inside this window the icons labeled (4) represent a subprogram that extracts from the user’s file text messages describing the particulars of that evanescent scan and two 2D arrays representing frequency shift and Q data vs. x-y sample location. This data is then formatted, x and y sample coordinate vectors are created, user saved content messages are displayed, and axis labels are applied to a LabVIEW subprogram that plots the 3D data sets as defined by the user interface. The data structure, which performs all of the 3D graph formatting, is known as a “sequence”. The above sequence contains 0-3 windows of graphical programming that execute sequentially. This type of structure allows the programmer to enforce the order of specific algorithmic steps. Contemporaneously with icon group (4), the icons labeled (5)
choose one of two strings to write to the variable “XY Units”. The contents of this variable are displayed on the front panel at the bottom of the graph. The choice of the two strings is made by a Boolean control that can be defined by the programmer before program execution. Note that the code is set to display the string “X and Y units = mm”, as can be seen at the bottom of the 3D plot on graphical user interface front panel.

Figure 4-6 illustrates the hierarchical structure of the microwave microscopy software. Each block represents a graphical software module. These modules are arranged in a tree to show how upper-level modules call individual sub-modules. For example, the top-level module labeled EMS 3 orchestrates the overall algorithm. All of the connected modules below it support and carry out the signal processing, data storage, data presentation, and communication requirements.
Figure 4-5. LabVIEW Display Generation Graphical Program
Figure 4-6. Microwave Microscope Algorithm Module Hierarchy
4-3 PCI Bus

The peripheral component interconnect (PCI) is a versatile processor-independent local bus structure, illustrated in Figure 4-7, that was introduced by Intel Corporation to enable CPU, memory, and peripheral device interconnections for peer-to-peer transfers of 64-bit words at up to 66 MHz rates, or 4 Gigabits per second, using burst packet transfers. A 32-bit bus and 33MHz transfer rates are automatically supported when slower devices are installed. None of the bus devices have dedicated memory or address assignments, but instead are configured and so assigned by BIOS flash memory on power up. Power conserving reflected-wave logic switching is employed that requires only one-half logic level voltage excitation without the requirement for bus line impedance termination, but bus lengths must be short. Bus bridge extenders are accordingly used between separate PCI bus segments, and to other buses such as ISA, which permit concurrent separate bus operations. Up to 256 PCI buses can be supported with bridges each with a maximum of 10 peripheral devices. CompactPCI is an industrially hardened modular PCI bus available in a 3U or 6U Eurocard form factor intended for embedded applications where robustness is essential. Implementations include communications servers, industrial automation, and defense electronic systems. The PCI bus typical bandwidth of 132 Mbytes per second supports high-speed applications such as video image manipulation, whereas the ISA bus bandwidth of 8 Mbytes per second cannot.
The PCI bus provides a contemporary solution for high-speed, custom-application, virtual instrument implementation.
Chapter 5: Analytical Instrumentation In Advanced Control

Analytical instrumentation is described for increasing the attribution of multisensor information systems featuring analytical ex situ planners applied to process control. Planners provide control advancement by assessing aggregated and evolving measurements during processing to implement a global process real-time quality control loop.

Analytical instrumentation for feature attribution often involves detailed multidimensional microstructure assessments, including process parameter pattern recognition comparisons with processing goals. For example, thin-film deposition processes require increased data attribution to describe crystalline growth mechanisms (bonds, evaporation, adsorption), physical properties (mass, phase, species), and structure (boundaries, geometry, morphology) in order to accurately define process actuation parameters (gas/liquid feedstock flows, heat, pressure). Implementation may be met, as illustrated in figure 5-1, employing a DSP accelerator and open-standard embedded VXI computers. VXI instrumentation backplanes permit interoperability between different vendor hardware and a mix of software systems in order to combine technologies ranging from discrete instruments to multifunctional virtual instruments.
Figure 5-1. Hyperspectral Analytical Instrumentation System
5-1 Autoassociative Crystal Feature Extraction

Following sensor data acquisition and virtual instrument feature identification a crystal facsimile classification is performed by means of an autoassociative network, described in figure 5-2 that has been trained on exemplars available from the Japan Science and Technology Corporation enabling reliable identification through crystal regularities. An autoassociative neural network can be likened to a content-addressable memory. For instance, such a network is trained so that output patterns are recalled from the input pattern (e.g. training-input pattern $A + \varepsilon$ recalls training-output pattern $A$, where $\varepsilon$ is a small perturbation pattern). The input and its associated output-training pattern are identical, hence the name autoassociator network. Such networks are inherently robust. They are designed to recognize correct output patterns given corrupted or incomplete input patterns. For this application the autoassociative network is a good choice because it exhibits the unique property of closed separation surfaces unlike typical multi-layer perceptron (MLP) classifiers, which have open separation surfaces. In other words, the autoassociative network establishes closed decision boundaries in the input space, which tend to capture the probability distributions of the training examples. Multi-layer perceptrons have no means for discriminating or indicating the confidence of which it classifies input patterns. That is, false patterns arbitrarily far away from the distribution of positive exemplars can be erroneously classified. Given this, autoassociative neural networks are prime
choice because they draw closed regions around the valid training data thus enabling classification decisions to be reliably made.

A tutorial is provided in Appendix III, which imparts more details describing autoassociative networks and their property of closed separation surfaces (a simulation is provided). It also presents, by way of a structure that augments the autoassociator, a useful method of controlling the separation surfaces for optimal classification discrimination.
Figure 5-3 illustrates the fundamental building block of all neural networks. Figure 5-4 describes details concerning an autoassociative neural network with a pyramidal structure. Equation 5-1 describes the computation of the network output.
Figure 5-4. Pyramidal Structure of the Autoassociative Neural Network
\[ y_j = f \left( \sum_{i=1}^{N} w_{ij} \left[ f \left( \sum_{k=1}^{M} \omega_{ki} x_k + \theta_i \right) + \phi_j \right] \right) \]

\[ f(u) = \frac{2}{1 - e^{-\alpha u}} - 1, \text{ Gain Factor } \alpha > 0 : \text{ Activation Function Definition.} \]

\[ \omega_{ki} = \text{Weight connection from input k to hidden neuron i.} \]

\[ w_{ij} = \text{Weight connection from hidden neuron i to output neuron j.} \]

\[ \theta_i = \text{Bias input of neuron i.} \]

\[ \phi_j = \text{Bias input of neuron j.} \]

\[ N = \text{Number of hidden neurons.} \]

\[ M = \text{Number of input/output neurons.} \]

\[ X(t) = \begin{bmatrix} x_1 \\ \vdots \\ x_M \end{bmatrix}, \quad Y(t) = \begin{bmatrix} y_1 \\ \vdots \\ y_M \end{bmatrix} \quad \text{where } x, y \in [-1,1] \]

\[ t = \text{Refers to the pattern number in the training set, } t \in \text{Positive Integers.} \]

Note: For autoassociative networks, the input/output training pairs are identical. That is, \[ X_T(t) = Y_T(t) \forall t. \] The subscript \( T \) denotes “training” vectors.
Real-time planner systems operate at a higher level of abstraction than the interfaced control algorithms that complete a processing system. Further, hyperspectral imaging is a versatile collateral methodology that integrates both spatially and spectrally continuous data simultaneously to assist product characterization. The attribution of a crystalline thin-film facsimile, from multisensor chemical composition and morphology structure virtual features, is expedited by DSP algorithms executing rapidly repeating nested multiply and accumulate operations exemplified by equation 5-1. Furthermore, this DSP platform performs all of the computations required by the autoassociative algorithm for hyperspectral data integration as well as the product-goal comparisons and controller references adjustments.

A VXI bus internally serves the VXI embedded-computer-backplane I/O with a GPIB interface for discrete instrument sensors. A local Ethernet link is provided for serving the host computer because of its 10/100 Mbps data rate and up to 1518-byte message size capability. Ethernet employs a CSMA/CD network access algorithm, which has negligible time delay and blocking when network traffic is restricted by limiting the number of connected nodes. Efficient wideband peer-to-peer digital interfacing up to 400 Mbps between the digital signal processing accelerator and VXI backplane is provided by an IEEE 1394 FireWire serial bus, originated by Apple Computer and also known as DigitalLink. FireWire supports both asynchronous and isochronous
data transfer with a six-wire cable. Its physical layer tree topology is automatically reconciled with each network node change without host computer intervention, but interconnection distances are presently limited to 5 meters. FireWire data transfers are memory-based rather than channel-addressed to enable efficient processor-to-memory CPU transactions.
Conclusions

This thesis has presented a comprehensive description for contemporary instrumentation delineated by function and application for use in process control systems. In particular, the taxonomy is beneficial for making prudent design decisions when outfitting the instrumentation at the environmental-, in situ-, and ex situ-levels of the hierarchical process control scheme presented in chapter 1. The usefulness of the taxonomy was justified through the development of a signal model, which aided the assembly of the instrument classification through its hierarchical structure. This classification was further validated by detailed design contribution examples each illustrating a level in the overall taxonomy.

The levels of the instrumentation taxonomy include:

- Analytical Instruments
- Virtual Instruments
- Discrete Instruments

With this instrument foundation, a framework for hyperspectral ex situ planner instrumentation was described for use in advanced material process control applications. An ex situ planner must perform near real-time microstructural interpretation and goal-product comparisons in order to continually guide the process parameters to achieve optimal product synthesis. The behavior required of ex situ
planners implies high-performance algorithms, which further implies the subsequent
need for high-performance instrumentation to integrate multisensor data for its
purpose. The ex situ planner must perform hyperspectral imaging in order to glean
information concerning the current state of the product with respect to the product
goal. The data processed from heterogeneous sensors containing both spatial and
spectral content allows the planner to capture information not available when
processing data from a single sensor. This level of data processing becomes very
useful because it allows for cognitive interpretation, which can be exploited through
symbolic methods (i.e. neural networks, etc.) for generating product facsimiles and/or
automatically adjusting process references.

Analytical instrumentation was described for the ex situ planner. This instrument
employed a DSP processor suitable for performing pattern recognition for feature
classification. It performs this task by implementing an autoassociative neural
network trained on prototypes encoded with crystalline feature data. The
autoassociator network was chosen for its discriminatory behavior (i.e. closed
decision boundaries). This property allows accurate feature recognition, thereby
providing reliable process control adjustments.
Bibliography


Bus Standards and Networking Architectures, [http://www.techfest.com](http://www.techfest.com)


Appendix I

The AC Magnetization Software Control Algorithm

1. The software user must set up the GPIB addresses of the Lakeshore temperature Controller and the Keithley DMMs as well as the temperature units (i.e. Kelvin, degrees Celsius, etc.), temperature resolution, heater current, curve number for the thermocouple device, etc. See Figure A-I-3 for the Magnetization Test Setup Window.

2. A temperature range must be specified as well as the number of data points to be taken. \(<\text{Tmin}>\), \(<\text{Tmax}>\), and \(<\#\text{of Points}>\).

3. The values of magnetic field amplitudes must be defined (up to 10 field values).
   The possible field range for the values is 0 to 3.6 Oe.

Once the above test parameters are specified and the test is initiated by the user, the data collection process begins. The following diagram illustrates a simplified version of the data collection procedure:
Figure A-I-1. The Data Collection Algorithm.

T[i] is an array of temperature values over the range from T_{min} to T_{max}. M[j] is an array of magnetic field amplitude values defined by the user.
Software Features

In the test setup window complete control setup is provided for Lakeshore temperature controller and the Keithley DMMs. A toggle switch specifies which DMM measures susceptibility and which measures loss.

The Measurement Control Window (see Figure A-I-4 – Measurement Control Window) provides graphs that are updated after each data point, a window of active controls, and a window containing indicators.

In the window of controls, the user can actively alter such things as the heat power percentage, the heater current, and the set point tolerance. The set point tolerance specifies a temperature discrepancy margin for which the program will take data. Data averaging has been provided in this program and a control box specifies how many data points to take and average for each temperature value. The control marked AutoPilot controls the thermal drift rate so that it remains relatively constant at a rate that allows data to be properly averaged and recorded (i.e. ~0.02 Kelvin/sec). It does this by automatically adjusting the heat power percentage. The Update Period control box adjusts the reaction time of the AutoPilot control. An abort button allows the current test procedure to be discontinued. Once the measurement procedure is finished or has been aborted the results can be saved to a file. The file is saved in a convenient ASCII text format. All of the key parameters as well as the collected data
for susceptance and loss are stored in an easy to read manner (refer to Figure A-I-6 – Sample Data File).

The window of indicators contains readouts of the current set point (or temperature value for the next data sample series - i.e. T[i] referring to Figure A-I-1) and the current sensor temperature.

Previously saved images can be viewed by selecting “Load Image” from the Initialization Window (see Figure A-I-2 – Initialization Window) and selecting the appropriate data file. Once the image is loaded, the graphs of susceptance and loss versus temperature and the ASCII text file can be inspected. The data may be then printed to either a local printer or to a file in the form of a JPEG image (see Figure A-I-5 – Print Setup Window). The print setup screen allows the user to print or create a JPEG image of a single graph, or both graphs on a single page.
Figure A-I-2. Initialization Window
Figure A-I-3. Magnetization Test Setup Window
Figure A-I-4. Measurement Control Window
Figure A-I-5. Print Setup Window
Magnetization Image Data - User Information...

Susceptibility Gain = 8
Loss Gain = 10
#of points Averaged = 5
Sample Resolution =  xxx.xx
Sample Units = (K)
Tmin. =  77.5000
Tmax. =  220.0000
# of Points =  75
Magnetic Field Amplitudes =
0.0250  0.1000  0.2500  0.5000  1.0000
Magnetization Test Mode = Thermal Drift
Keithley 175 Measures -> (Susceptance)
Keithley 197 Measures -> (Loss)
Set Point Tolerance (+/-) =  0.2000
Proportional (Gain) =  1.0000
Integral (Reset) =  0.0000
Derivative (Rate) =  0.0000

Susceptance Data:
11.742811.768511.745211.724211.6903
11.760611.790711.769611.733911.7028
11.749711.781611.759711.724311.6934
11.739711.770211.748511.712611.6802
11.735611.746711.716311.684811.6526

Loss Data:

Temperature Data:
77.590077.592077.614077.648077.6720
79.552079.794080.074080.302080.5940
81.410081.722081.964082.224082.4920
83.202083.126083.054083.092083.1840
85.204085.394085.572085.746085.9060
87.086087.220087.268087.502087.6340

Figure A-I-6. Sample Data File
Appendix II

The Microwave Microscopy Software Data Acquisition Algorithm

Pre-Scan Initialization Procedure:

1. Adjust the vertical positioning of the EMP.
2. Run preliminary platform leveling setup.
3. Move the XY platform to the initial sample point.
4. Set the microwave source frequency to $f_0$ (the unloaded resonant frequency of the probe) by setting the DC offset of the 1F signal.
5. Set the modulation width, or the sweep of the frequency modulated EMP excitation signal. Adjusting the amplitude of the 1F signal sets this attribute.
6. Set the XY dimensions of the sample under test.
7. Set the sample frequency, $f_s$, for the data acquisition devices.
8. Set the modulation frequency (i.e. how fast the excitation signal will sweep through the range of frequencies).
9. Set the modulation sensitivity (MHz/V). This is the ratio of how many megahertz the excitation signal will deviate from the center frequency (set in step 2) per volt at the microwave generator’s control input. In other words, it lets the software know how the voltage of the 1F signal translates into the excitation signal frequency.
10. Set the stepper motor resolution (micron/step).
11. Set the RF power level (dBm).
Automated Scanning Procedure:

1. Move the XY platform to the next sample point.

2. Measure both the 1F signal and the resulting 2F signal. The 2F signal is a measurement of the reflected signal’s amplitude corresponding to the frequencies swept by the 1F signal. Obtain 5 cycles of the periodic waveforms. Maintain an integer number of cycles to ensure the integrity of the following calculation.

3. Apply a high-pass filter based on a variation of a moving average filter to remove low frequency components from the acquired signals.

4. Evaluate the discrete time Fourier series (DTFS) amplitude at \( f \) for the 1F signal and at \( 2f \) for the 2F signal. The amplitude is computed using the following equation. \( f_m \) is the harmonic frequency at which the amplitude is desired. \( f_s \) is the sampling frequency. \( N \) is the length of the acquired signal. \( x \) is the vector of signal samples.

\[
M(f_m, x) = \frac{2}{N} \sqrt{\sum_{i} x_i \cdot \sin\left(\frac{2\pi \cdot i \cdot f_m}{f_s}\right) + \sum_{i} x_i \cdot \cos\left(\frac{2\pi \cdot i \cdot f_m}{f_s}\right)}
\]

5. Use feedback control (a PI loop controlling the DC offset of the 1F signal) to move the center frequency so that the 2F signal aligns symmetrically on the new resonant frequency due to the influence of test material on the EMP. When the center frequency (the dc offset of the 1F signal) converges to the new resonant frequency the 2F signal will contain mostly \( 2f \) spectral components. The error signal of the PI
controller (ε) will become small as |2F|_{2f} (defined as the 2f amplitude contained in the 2F signal) becomes large. When the error signal shrinks within a preprogrammed dead band region the feedback control terminates and the new resonant frequency is said to have converged. For this, a frequency shift value is stored (the DC offset of the 1F signal modified by the modulation sensitivity scaling factor to convert Volts to MHz). The current 2f amplitude of the 2F signal (i.e. |2F|_{2f}) is also stored; this value is a direct-proportional measure of the quality factor associated with the new resonant frequency. Note that the units are in Volts.

6. For each 2-dimensional point over the material sample, values of frequency shift and Q are compiled and plotted.
Appendix III

Introductory Tutorial of Autoassociative Neural Networks

Introduction

A difficulty with many classification architectures is that they produce an output for any input. Training data may only be in certain regions of the input space. The classification model will produce an output for any given input in the input space. Decisions will be made in areas arbitrarily far away from the training set. For this reason, there needs to exist a mechanism for the model to indicate the confidence of its decision. A property of autoassociator networks is the closed separation surfaces they establish in the input space. This section explores the possibility of using such networks to draw closed regions around the valid training data so that classification decisions can be reliably made. A simple system is simulated to illustrate its usefulness.
Problem Description

An issue with multi-layer perceptron (MLP) classifiers is that the separation surfaces they form are likely to be open. These feed forward networks are likely to partition the input training data properly, but the surfaces are not likely to be closed. That is, they do not envelope the data examples by capturing their probability distributions. The problem that exists is due to the open separation surfaces created by the learning algorithm. False patterns that are far away from the distribution of positive exemplars can be erroneously classified as being in a particular class. As it stands the MLP has no means for discriminating or indicating the confidence of which it classifies input patterns. What is needed is a way to specify a region around the positive data so that if the input pattern is within some distance from these exemplars then the classification confidence is high. That is, if the input data is within this region the MLP classification is assumed to be valid otherwise the confidence is deemed low.

The network topologies used in this paper follow the general pyramidal structure depicted below. The input pattern is designated by $X_o(t)$ and the output pattern is indicated by $X_l(t)$. The ‘t’ refers to the pattern number in the training set. All neurons in the hidden layer and output layer have bipolar sigmoidal activation functions. This “squashing” function binds the output to the closed interval $[1, -1]$. 

The figure A-III-2 demonstrates the open separation surfaces generated by an MLP classifier. The training data consists of eight points on a circle. Different points on the circle are classified into separate groups indicated by the color in the centers of their data point. The whole input space was sampled and the results were classified and indicated according to color. The spatial areas not corresponding to a valid classification pattern were left white. The valid classification patterns for this 2-D system were \{(1,1), (1,-1), (-1,1), (-1,-1)\}. The MLP classifier hidden layer consisted of two neurons.
This pictorial clearly demonstrates the open separation surfaces generated by the classifier and the possible problems that arise due to these large open surfaces. If the objective is to classify points corresponding to a fixed circle then it is apparent from this illustration that data points far away from the circle data can also be classified. This is the dilemma in clear terms.

Figure A-III-2. The Open Separation Surfaces Created by an MLP Classifier
A Possible Solution

It would be desirable to superimpose closed separation surfaces upon the MLP classifier to maintain model/classification integrity. A possible realization of this idea involves a structure using an autoassociator neural network.

An autoassociator is a neural network trained such that the output is forced to reproduce certain input patterns. It is well known that feedforward autoassociators always exhibit closed separation surfaces (note: they have the same feedforward pyramidal structure as the classifier).

An additional criterion is used in conjunction with the autoassociator. The Euclidean input-output distance is taken such that if it is less than some delta, then the pattern is accepted otherwise it is rejected:

\[ \left\| X_L(t) - X_0(t) \right\|_2 < \delta \]

The idea is that only the patterns used for training the autoassociator are likely to be reproduced with enough approximation at the output. All other patterns including the false ones are not likely to be reproduced with the necessary approximation.
Figure A-III-3 shows the proposed system. The idea is to augment the MLP classifier with the structure pictured below. Essentially, the classifier and this autoassociator network will operate in parallel. The classifier will classify all input patterns with the open separation surfaces while the autoassociator network indicates which output patterns should be considered reliable. In effect, the autoassociator “masks” the input space where classification is valid.

The autoassociator is trained on all the positive training exemplars used in the MLP classification training process. The term “positive” refers to those patterns in the classification that are valid (i.e. the eight points on the circle). During operation, each input pattern goes to both the classifier and the autoassociator. If the input pattern autoassociates then the input/output distance should be very small. If this distance is less than some $\delta$ then the pattern is accepted and the MLP classification decision is considered valid otherwise the decision is given with low confidence.
The Simulation Methodology

To illustrate the proposed methods, an autoassociator network (as shown above) was simulated for the classification of points on a circle. The autoassociator was trained using typical back-propagation learning based on gradient descent optimization. The cost function is given as:

\[ J = \frac{1}{2} \sum_{t=1}^{T} \sum_{j=1}^{N} \left[ d_j(t) - x_j(t) \right]^2 \]
Where,

d_j(t) = The j^{th} element of the desired output pattern indexed by \( t \).

x_j(t) = The j^{th} output element of the network excited by an input pattern indexed by \( t \).

N = The number of output neurons.

T = The number of patterns to recall.

The entire simulation was coded using C++. The graphics were generated from the simulation data and plotted using MATLAB.

Results

Figure A-III-4 shows the closed separation surfaces generated by the autoassociator network for various values of \( \delta \). Note how the choice of \( \delta \) will allow the classifier to generalize over a particular area. A large value of \( \delta \) will result in large autoassociated areas and smaller values of \( \delta \) will constrict these areas around the data series. The choice of course depends on the particular application of interest.
It is noteworthy to point out the small region centered at the origin (0,0). Because this represents patterns with very small elements (i.e. nearly zero) it is expected that the output of the network will also be near zero, as long as the bias values of the neurons are sufficiently small. Due to this the input/output Euclidean distance will also be small. Hence the patterns near zero are considered valid by the autoassociator model.
Learning from positive data only can cause over-training. This can sometimes result in a large auto-associated area or the inclusion of unwanted patterns. By introducing negative examples, the autoassociator can learn to exclude certain undesirable patterns. To a certain degree, the autoassociated areas can be shaped in this way. To do this a penalty function must be added to the training cost function to provide a means for restricting the growth of the closed surface in the pattern space. This was not done in this simulation but it is possible nevertheless.

The following two figures are examples showing the superimposed closed separation surfaces upon the MLP classifier input space for two different $\delta$ values, 0.2 and 0.3 respectively. In the first figure ($\delta=0.2$) the magenta regions envelope the circle data and each region does not overlap any of the neighboring classes. In other words, it is visually apparent that the network would perfectly classify any data falling in those closed regions. In the figure with $\delta=0.3$ the yellow regions represent the closed separation surfaces. Again the autoassociator performs perfectly except for the right most autoassociated area slightly overlaps the input space classified as “blue” (class B). This blue area is not part of any of the classified data. This result could pose a problem but can easily sidestepped by adjusting $\delta$. 
Figure A-III-5. Closed Separation Surfaces With $\delta=0.2$

Figure A-III-6. Closed Separation Surfaces With $\delta=0.3$
Conclusions

Depending on the actual application this model provides a very nice way to establish reasonable bounds for which the output of an MLP classifier can be considered meaningful. Large values of $\delta$ allow the classifier to generalize over a larger area. This can sometimes cause false patterns to be wrongly accepted. Different applications will require different $\delta$ values. $\delta$ is a design parameter that is generally varied so that the output error, or the classifier’s discrimination ability, is within a desired bound.