The Virtual Manufacturing Cell

Charles R. McLean
Howard M. Bloom
Theodore H. Hopp

Factory Automation Systems Division
Center for Manufacturing Engineering
National Bureau of Standards
Gaithersburg, MD 20899

Presented By:

Charles R. McLean
Fourth IFAC/IFIP Conference on Information Control Problems in Manufacturing Technology
Gaithersburg, Maryland
October 1982

Bibliographic Reference:

THE VIRTUAL MANUFACTURING CELL

C. E. McLean; L.N. Bloom; and T. H. Hopf

Industrial Systems Division
Center for Manufacturing Engineering
National Bureau of Standards
Washington, D.C.

Abstract. A virtual manufacturing cell is being developed at the National Bureau of Standards as part of the control software for the Automated Manufacturing Research Facility (AMRF) project. The traditional group technology (GT) cell has evolved from the need to maintain the flexibility to manufacture a family of parts while gaining some of the efficiency associated with a single process flow line. GT cells are normally defined by a fixed physical grouping of machining workstations that produce a particular class of parts. A shop based upon virtual manufacturing cells provides greater flexibility than existing GT shop configurations by time sharing machining workstations. Virtual GT cells are not identifiable as fixed physical groupings of machinery, but as data files and processes in a control computer. Functions performed by these processes include analysis, reporting, routing, scheduling, dispatching, and monitoring. At a higher level, the shop control system schedules cell activation and allocates workstations and other resources to these cells. Workstations are at all times under the control of either a particular virtual cell or a pool cell composed of idle workstations.

Keywords. Production control; Hierarchical systems; Group technology; Artificial intelligence; Manufacturing processes; Management systems.

INTRODUCTION

The AMRF Project

A new type of Group Technology (GT) manufacturing cell, called a virtual cell, is being developed at the National Bureau of Standards to address specific control problems encountered in the design phase of the Automated Manufacturing Research Facility (AMRF). The project is investigating the automated production of small batches of machined parts. A portion of the NBS Fabrication Technology Division machine shop is being converted to a small testbed system that will be used for experiments in precision machining, automated process metrology, and manufacturing interface standards. For further information on the project, see Simpson, Meken, and Albus (1982).

Implementation Techniques

This section identifies control and data processing methodologies that will be employed in the construction of the virtual cell. These techniques have been selected because they appear to provide the greatest overall system reliability and potential for real-time adaptive control. Detailed discussions of most of these techniques can be found in other NBS papers: Albus (1981); Albus, Barbera, and Nagel (1981); Albus, colleagues (1982); Barbera, Fitzgerald, Albus (1982).

Hierarchical control. This organization is equivalent to the line or tree structure found in many conventional manufacturing systems. Each system takes commands from only one higher level system, but may direct several others at the next lower level. Long range goals or tasks enter the system at the highest level and are decomposed into sequences of subtasks to be executed as procedures at that level, or output as commands to the next lower level. Guidelines for the design and implementation of hierarchical, multi-level systems can be found in Nezarovic, Mack, and Takahara (1970).

Local intelligence. At each level in the control hierarchy this processing capability enables the system to decompose tasks, analyze feedback, and respond to problems at that level. It also ensures that only major tasks, having a global impact, will be handled by the decision making systems at the higher control levels. Guidelines for using local intelligence in the automation of managerial control can be found in Beer (1982).

State machines. To ensure that the control system is deterministic, it will be defined as a network of state machines. All inputs,
outputs, states, and state transitions of the system are identified in a state graph. The graph is used to define state or decision tables which are processed by the control system. For information on the implementation of state or decision table based systems, see Metzner (1977).

Control cycle. A time interval, called a control cycle, is defined for each control subsystem; this cycle determines how often its state table is processed. Processing a state table involves sampling state variables, searching the table for a state that matches the sampled variables, executing the routines, and generating the outputs that are associated with the selected state. The cycle at each level must be short enough to maintain stability; the processor must be able to identify the current state and generate appropriate outputs before the behavior of the system deviates from acceptable ranges.

Planning horizon. The amount of time that any system plans into the future to perform the tasks at its control level is defined as its planning horizon. It is determined by the tasks or goals that are passed down as commands from the next higher level. Systems do not know about events or activities which will occur beyond their planning horizon. In general, a system cannot plan beyond its current command or goal for it does not know what the next command may be. By defining shorter and shorter planning horizons at each successively lower control level, the processing capacity required for planning during control cycle is kept to a minimum at every level.

Hierarchical scheduling. This technique, the partitioning of activities or jobs by large time increments at the higher levels and smaller increments at the lower levels, frees each control system to make the decisions at its level that are necessary for efficient operations. For example, a higher level may schedule by grouping jobs into partitions or packets by the month that the job is to be performed. The jobs in Packet #1, are accomplished in the first month; Packet #2, the next month, etc. The next lower control level is only tasked with the jobs in Packet #1, thus limiting its planning horizon to the current month. It divides the jobs into packets by week into Packet #2.1, 2.2, etc. The next lower level, with a one week planning horizon, would be tasked with packets #1.1.

Since the hierarchical control structure requires that commands must flow downward, lower level systems cannot by themselves move a job out of the packet given to them by a higher level. If the job cannot be processed in the specified time frame, the reason for the failure must be reported as feedback to the next higher level. The controlling system at the next level may then take action to either circumvent the failure or reschedule the job by tasking the subordinate system with a modified packet of jobs.

Communications by common memory. All systems will communicate by passing messages through mailboxes in a common memory or data base. Each system will have a command mailbox, where its controlling system can write commands, sensory mailboxes for processed sensory data, and status mailboxes for feedback from controlled systems. Each mailbox can be written to by only one system, but can be read by any other system. The mailboxes, updated every control cycle as a part of the state machine implementation, also provide a snapshot of the current state of the system, useful for diagnostic analyses and system restarts.

The AME Control Hierarchy

The overall control structure was developed from an in-depth functional requirements analysis of conventional manufacturing management systems. These organizations perform the tasks necessary for the planning and control of production while often incorporating techniques that are analogous to those discussed above. The planning and decision making functions in these non-automated manufacturing systems are distributed among a hierarchy of employees, thus permitting a high degree of parallel processing (necessary to most real-time adaptively controlled systems). Further analysis of functions performed in both automated and non-automated manufacturing control systems are described in Groover (1980), Bjorke (1981), Chase and Aquilano (1977), Halevi (1980), and Kochhar (1979).

This requirements analysis has resulted in a design for the AME hierarchical control system that is composed of five major levels (Fig. 1): Facility, Shop, Cell, Workstation, and Equipment. Each major level is further decomposed into sublevels or modules as described below.

Facility control. At this highest level of control, there are three major modules: Manufacturing Engineering, Information Management, and Production Management. Manufacturing Engineering provides user interfaces for the design of parts, tools, fixtures, and for process planning. Information Management performs cost and inventory accounting, customer order handling, and procurement functions. Production Management generates long range schedules and production planning data used for tasking and managing the shop control system at the next lower level.

Shop control. This system is responsible for the real-time management of resources and jobs within the shop through the three major modules: Task Management and Resource Allocation. The first schedules job orders, equipment maintenance, and shop support services, such as housekeeping. The latter
allocates workstations, storage areas, tools, and material.

Cell control. The sequencing of a batch of jobs through workstations and the supervision of various support services, such as material handling or calibration, is managed at this level. Modules exist at this level to perform analysis, reporting, routing, scheduling, dispatching and monitoring.

Workstation control. The activities of small integrated groupings of shop floor equipment are directed and coordinated at this level. A typical workstation, consisting of a robot, a machine tool, a material storage buffer, and a control computer, processes a tray of parts that have been delivered by the material handling system. The control modules sequence equipment level subsystems through setup, cutting, chip removal, in-process inspection, teardown, and cleanup operations.

Equipment control. These controllers can be identified with particular pieces of equipment on the shop floor, such as robots, machine tools, coordinate measuring machines, carts, carousels, and various storage-retrieval devices. Standardized interfaces will be developed, as necessary, for commercial production equipment to provide compatibility with EHS workstation level controllers.

THE STRUCTURE OF THE CELL

Evolutionary Trends

The definition of the GT cell concept may be just the first stage of an evolutionary process (Fig. 2), the development of a flexible material processing system around a family of parts. Stages in this evolution include production by part family, automation of processing equipment, virtualization of control structures, and incorporation of machine intelligence.

Part family production. The use of cell structures and task decomposition seems to be a reasonable solution to the complex job shop management problem. Families are defined using a classification scheme that groups parts according to processing requirements, geometric shapes, tools used, production costs, and/or material composition.

The part family associated with a particular GT cell is normally determined by similarity of processing requirements. The cell brings some of the efficiency of a flow shop to small batch production by using a set of machine tools and shared job setups to produce that part family. Group technology theory and implementations are discussed in detail in Hyde (1981), Desai (1981), and Goveer (1980).

Automation. Although it is not essential to the implementation of the GT concept, most cells currently have some level of automation. The increased performance of operator and supervisory functions by computers, numerically controlled tools, robots, and material handling systems can be viewed as the second stage of cell evolution. A demand for standardized interfaces has resulted during this stage from the industry’s need to construct integrated automated systems with equipment procured from different manufacturers.

Virtualization. In this third stage of development, dynamic control structures are introduced. The cell is no longer identified with a fixed set of workstations or equipment on the shop floor. Access to workstations is time shared among cell level controllers by a method that is similar to the central processing unit (CPU) time sharing that is used in many computer operating systems.

Machine intelligence. The final stage of cell development will involve the incorporation of machine or artificial intelligence. The sophistication of the system is increased by adding capabilities to generate complex plans with alternative courses of action, evaluate and optimize these plans, learn from experience, and reorganize its structure to use learned techniques to solve problems in new ways.
The Group Technology Cell

In a shop based on GT cells, it can be assumed that orders have been screened and that only the orders for the part family of a particular cell are passed to that cell for processing. AMF researchers have identified the following major functions as being necessary to the management of a GT cell: analysis, reporting, routing, scheduling, dispatching, and monitoring. In conventional (non-automated) job shops, these functions may have been performed within the context of the cell either by supervisory personnel, or in some cases, at a higher level outside of the shop by programs in a central computer.

Analysis. This function identifies the job to be done, decides whether or not it can be done, and determines the constraints that apply or the efficiencies that may be had by performing certain jobs in conjunction with others. Large packets or batches of jobs assigned to the cell are decomposed into smaller sub-batches.

Questions which must be addressed by analysis include: (1) Is this a normal production or an initial prototype run (process plan verification)? Procedures will probably be different in each case. (2) What processing capabilities will be required to produce the batch? (3) What special tools, fixtures, and materials will be needed? (4) How long will each individual operation be expected to take? (5) What quality assurance measures must be employed to ensure that the batch of parts meets established tolerances? (6) Which parts of the batch order are critical, and which may be delayed if necessary? (7) If any subsystem fails, what corrective actions must be taken? This is only a partial list, many additional analysis problems for the cell can be envisioned.

Routing. This function selects the appropriate sequence of workstations to perform the operations on a part or a batch of parts. Accurate time estimates for the processing at each station must be generated. The routing function must also identify the additional resources, such as storage space, fixtures, grippers, probes, and tools that will be required at each station. If the batch order gets behind schedule, decisions to use splitting or overlapping techniques may be made to increase processing capabilities and ensure its timely completion.

Scheduling. This function determines the actual clock times that workstations will be required by the cell and when major activities will begin and end. There may be several workstations of the same type on the shop floor; scheduling will identify precisely which ones will perform the actual machining, handling or inspection processes.

Dispatching. This function uses the schedule to initiate and coordinate workstation level operations. Dispatch orders are used to initiate the loading of material from inventory into trays, the movement of trays of parts, tools, fixtures and grippers between stations, the downloading of process plans and job related data, the processing of parts at a station, and the performance of cleanup or housekeeping operations after processing is completed.

Fig. 2. The evolution of the manufacturing-cell
Thohiemtructumforrobotcontrol
doroloprdat;raSprovideareliable,real-
timerocontrolthroughtrokdocorpaotion,
dlatributedprocerring,andartriticor
fixedprocessingotructure.Althoughthia
architocturoiradoquatefor8robot,there
toticcontrolrtruetureaeo8ed
inrpproprirtoforbieherhve188nrgement
rystomo,mucha0theeoll,thatrupervire
projrctrofvaryingritoAndcomplexity.A
dynamic,hiorarchicalstructureraa
onvirionedtbatcouldacquiremdn1bqui.h
oubryotemronLD.AS-neoded-is.
Thirnortypeofproductionoontrol
rtructurrvaaaamodthe'virtual'ce?lto
dirtinguirhitfromproviourreal-
murufacturin4colla,whichwerodofinedb
diportionsof.hopflooroquipacnt.me
virtualeo11preoontathemanufacturing
contra1Oy8teSraththoillUSiOnOf8
permanent.etofaosignedrorourcosinthe
oa~emannortbateconputorrithvirtual
memoryprrreotoit.uaorrwiththe
oppearas~oofalu&eaddreorrpco.
Thovirtualcellortodsthoconcoptofthe
cellbyoilouingthotlmerhrringof
Yorketatlomrlthotbarrituale0118that
producediflcrcntpartf~oilioa,buthavo
overlappingreaourcorrguirommto.Xnthis
evolution~ry.top,thecollis80longer
ldentifiatle00aIlxedphyrica:groupof
8acYnes.butnthera0acocputorproceu.
aaoocirteddetoandadyMmica11ycbang*
mtofworrrta:ioroontborhopf:oor(Pi(.
3a-g)-.
Addition.?function.,notnornall7
oraoci~tedri:hthoCTcell.aroroqsiredto
providethogrrrterf2exibili:yudbetter
nt1:rtiunafrnopro8oureoaproairedby
thiBarc2itecturr.Tbevirtualcellruat
incoryrrtenewcapeabilitieeforpredictiog
neodr,repaasltioatngreoourtee,time
sharingtheseeresourcrr,andtandrbakng
dulxhendoff.ofcont.roEedrubeyatem&
Psedicti=q.Sine.VOrkOta%On8andotber
re sour:^^armnolondrr:ermaaertly
roeigsed.tbecc:?corrrclier arnew
?rrdi:trbcnitrillacedaparrr:uiar
raeour:e.howlongitri:lboneeCod.AZ?
whatthooffrctorl?lbeofhavzto
itsibquiohitbrlo?~Utebatchi.con>let.d.
Ledtimerloractfvariearuaa0maror-rl
hndlsn&nurtbepredictedtorvoidhaving
aarlgneivorkrtationraftidleouait2rg
5eqnirltlonin~.Vhenavirtual0.11ia
er0~t.dbxtherhop,itrillnoraallyhave
a0r08lgnedrerourcea.Thejobprckrt
masignedbytheahopia8nalysedanda
atrateredevalopodforbcconplimhh#the
Jobotderarithinthbtpacket.Yprkatatioa8,
atoragempaee,'cuttarr.tray.,andotbot
rorourcorarethenrequi8ition.dfromtam
.hop.IfthedarirodreaourterareIngreat
aewaad.afterortire8trateair8and
rquiritionrrqkgsnerbtbd.

Requisitioning. When a virtual cell is
created by the shop, it will normally have
no assigned resources. The job packet
assigned by the shop is analyzed and a
strategy is developed for accomplishing
the job orders within that packet. Workstations,
storage space, cutters, trays, and other
resources are then requisitioned from the
shop. If the desired resources are in great
demand, alternative strategies and
requisitions may be generated.

Time sharing. Although this time sharing is
similar to that found in computer systems,
there are important differences. First, a
cell's time slice or access to a resource
will normally range from several hours up to
several days, rather than milliseconds as in
a computer operating system. Second, the
interrupt point at which a part may be
allowed to be removed from a machine tool
(due to a change in production priorities)
and placed in temporary storage will be
based on requirements of the particular
machining, assembly, or inspection process.
There will always be a cost associated with
interrupting a part's processing, normally
lost setup time. Third, a cell may not gain
complete ownership of a workstation area
during its access period. Sections of local
storage buffers at the workstation (including
the magazine on a machine tool)
may be allocated by the shop to another cell
to allow its work in progress to remain
at the station.

Handshaking. Protocols and procedures for
accepting and relinquishing control of
resources must be established. A handshaking
mechanism will be implemented to maintain
positive control of the resources during
changes in controlling authority. Rules must
be established governing the condition that
resources are to be left in when they are
relinquished. Common standard software and
hardware interfaces will be required due to
the increased interaction between a variety
of processes and processors in this dynamic
control architecture.

The Intelligent Cell

The final stage in the evolution of the
virtual cell is the extension of its local
intelligence. An expert system capability is
envisioned that would allow supervisory
personnel to incorporate their own knowledge
and experience into the system. Management
requirements for cells in different
manufacturing installations will vary, so
future cell control software will have to be
tailored to handle the management problems
and policies at each installation. More
sophisticated behavior can be provided by
improving planning capabilities, the
abilities to manage faults or crises; and to
learn from the same.

Conventional manufacturing systems may
Cell Operation Cycle
Phase 1

Idle shop, no virtual cells in existence, and all workstations under control of Workstation Pool Cell.

Cell Operation Cycle
Phase 2

Two virtual cells created, both cells require Workstation 4. Cell 1 has highest job priority.

Cell Operation Cycle
Phase 3


Cell Operation Cycle
Phase 4

Cell 1 begins overlapping batch processing at Workstation 2.

Cell Operation Cycle
Phase 5

Cell 1 releases Workstation 4. Shop Control reassigns it to Cell 2.

Cell Operation Cycle
Phase 6

Cell 1 completes batch, returns Workstation 2 to the Pool Cell, and is removed from the control structure.
exhibit very sophisticated adaptive behavior. If automated systems are to equal or surpass the sophistication of conventional systems, their design must facilitate the incorporation of new knowledge into the decision processes of the system. The behavior of the cell is described in terms of classes of intelligence that it may demonstrate (from lowest to highest): reaction, planning, optimisation, learning and self-organisation. Any cell implementation will probably combine features of several of these classes, but the first cell built for the AMF will concentrate on reaction and planning. A knowledge base will be developed that will allow the incorporation of the higher classes at a later time.

Reaction. The initial implementation of the cell will use simple decision tables and procedure calls to determine the state of the system and to look up or compute the appropriate responses for each state. This "reaction" class of behavior will be very simple, corresponding to that demonstrated by simple organisms.

Planning. This second class of intelligence incorporates the ability to look ahead or predict possible intermediate future states of the system and its environment (from the current state) and generate outputs that will take the system from the current state to the goal state. Planning procedure calls will be invoked from state tables. The size of the search space of potential solution paths leading from the current state through planned or predicted intermediate states to the desired goal state will be limited by the application of heuristics. Architectures for the construction of intelligent planning systems, called "pattern-directed inference systems", are discussed in Waterman and Hayes-Roth (1978).

Optimisation. The third class of intelligence, optimisation, will not be included in the early versions of the cell as a high degree of operating efficiency will not be critical in a research environment. When optimisation is implemented, planning procedures invoked from the state tables will be made more sophisticated through the addition of evaluation and simulation capabilities. Many alternative solution paths leading to a particular goal state could be generated. Simulations would be run which would permit the control system to evaluate the sensitivity of each potential solution to unknown random influences in its environment. The best path to the goal state would then be selected from an evaluation of simulation results.

Learning and self-organisation. The fourth class of intelligence includes learning, adaptation, and self-organisation. This behavior would permit a system to incorporate new data and procedures into its knowledge base from first-hand experience. It may be many years before this level of sophistication is effectively implemented in automated control systems. The learning process requires that significant experiences, data, or generated plans be recognised, and the procedures exist for incorporating this new information into the control structure. Adaptation and self-organisation capabilities would permit the system to change its own control structure and learning strategies.

Knowledge base. The implementation of an intelligent cell will require the development of a knowledge base that includes data about current tasks, production procedures, and the work environment. Waterman and Hayes-Roth (1978) have decomposed this data base into quiescent knowledge, active problem knowledge, and metaknowledge. Quiescent knowledge is general patterns, facts, and strategies relating to a particular problem domain. Active problem knowledge includes relevant rules and assertions which are applicable to the current problem at hand. Metaknowledge is comprised of rules for activating and acquiring knowledge, and for focusing attention during problem solving. Production-related data that may be found in the knowledge base includes job orders, production process and system models, schedules, etc.

The AMF Cell Architecture

The functions described above are divided into three hierarchical levels within the cell (Fig. 4): (1) task analysis and reporting, (2) routing and scheduling, (3) dispatching and monitoring.

Level 1. The highest control level within the cell, task analysis and reporting, is responsible for interpreting the commands from the shop control level and for
reporting status back to that level. The
commands from the shop will affect the
makeup of the packet of jobs and resources
assigned to the cell. Feedback reports will
include the progress of jobs, resource
requirements, equipment status, etc. The
analysis function will define sub-batches
and generate constraint information which
will be used by the routing and scheduling
level. Constraint information will acquire
various processing options which may affect
processing time, costs, or part quality.

Level 2. This control level will use the
sub-batches, constraints, and options output
from level 1 to generate routings. The
routing indicates types of workstations
required, the order in which they will be
visited, and the length of time the batch
will be expected to remain at each station.
Predicitive capabilities will be required to
estimate the duration of activities in order
to compute lead times for material handling,
etc. The scheduling function determines the
actual time when each resource or
workstation will be required and when it
will be returned to the shop. Tentative
schedules and resource requirements are
interpreted by the reporting function at the
higher level as feedback status. The
reporting system then attempts to acquire
the resources at scheduled times, or orders
changes to the schedule as dictated by the
availability of resources provided by the
shop.

Level 3. The lowest level uses the schedule
to perform dispatching, the formatting and
issuing of orders to workstation level
systems to move, process, assemble, inspect
or store materials and support equipment
(fixtures, tools, probes, grippers, etc.).
The monitoring function tracks the progress
of dispatch orders by interpreting the
feedback and status information of the
workstation level subsystems. Handshaking
protocols and standard workstation
interfaces are implemented at this level
within the cell.

CONCLUSIONS

There will undoubtedly be problems
associated with this new approach to
total: 1) The variable assignment of
workstations to a cell will require more
sophisticated and more flexible material
handling capabilities. 2) Longer distances
may be involved in the transportation of
trays between available workstations. 3) A
more complete knowledge of processing
requirements and system capabilities will
be needed to effectively manage this control
structure. 4) A number of virtual cells
competing for the same resources may produce
an undesirable behavior, similar to a
phenomenon called "thrashing" in computer
operating systems. A balance between virtual
cells and real processing capabilities must
be achieved.

There appear to be many advantages to
pursuing this approach to production
control: 1) Better utilization of resources
can be had through time sharing. 2) Cells
can be made to expand to handle increased
workloads. 3) The part family-based control
system provides a reasonable decomposition
of the overall production management problem
that a system programmer or designer can
readily understand. The designer can treat
the factory as a resource pool, and request
only needed capabilities. The shop control
system can present the designer with a
virtual factory, the illusion of unlimited
copies of dedicated resources. 4) The
technology associated with dynamic control
structures should be transferable to other
management or control problem domains,
such as construction, distribution and military
systems.

The evolving concept of the virtual cell as
dynamic manufacturing control structure
will undoubtedly undergo many changes as
development proceeds and it becomes better
understood. Several new production
management research areas, that will be
discussed in future papers, have been
identified as a result of this work.

ACKNOWLEDGMENT

The authors would like to thank Mary
Mitchell for her many critical reviews of
this paper.

REFERENCES

Robots. Byte Books, Peterborough, N.H.

Albus, J.S., Barbera, A.J., Nagel, R.H.
(1981). Theory and Practice of
Hierarchical Control. Proc. Twenty Third
IEEE Computer Society International

Albus, J.S., McLean, C.R., Barbera, A.J.,
for Sensory-Interactive Control of
Robots in a Manufacturing Facility.
Proc. Fourth IFAC/IIFP Symposium on
Information Control Problems in

Barbera, A.J., Fitzgerald, M.L., Albus,
Sensory Interactive Control System
Architecture. Proc. of the Fourteenth
Southeastern Symposium on System
Theory, April 1982.

Technology and Management Systems.
Proc. of the IFTP TC-5 International
Symposium on Automation of Production
Planning and
Control. Trondheim, Norway, 5-5
September 1980, North-Holland,
Amsterdam, 15-34.

Beer, Stafford (1982). The Shorter Catechism
of Stafford Beer. Derivation. 146-155.


This is to certify that this article was prepared by U.S. Government employees as part of their official duties and is therefore a work of the U.S. Government and not subject to copyright.

Shop Level

Analysis
Reporting

Routing
Scheduling

Feedback

Commands

Dispatching
Monitoring

Cell Level

Workstation Level

Fig. 4. Control levels with a cell controller.