J. Indian Inst. Sci., Sept.–Oct. 2001, **81**, 585–610.

IISc THESES ABSTRACTS

Thesis Abstract M.Sc. (Engng)

Evaluation of register allocation and instruction scheduling methods in multiple issue processors by Madhavi Gopal Valluri
Research supervisor: Dr R. Govindarajan
Department: Supercomputer Education and Research Centre

1. Introduction

Exploiting greater instruction-level parallelism (ILP) through multiple instruction issue and execution has gained significant importance in modern processors for achieving higher performance. Compilation techniques can analyze the program to expose parallelism, transform the program to enhance the parallelism, and schedule the program to exploit parallelism. This thesis focuses on two important ILP completion techniques, namely *register allocation* and *instruction-scheduling*. Register allocation is a technique using which frequently used variables in a program are kept in registers in order to reduce memory references. Instruction scheduling is a technique that reorders instructions to exploit paralleslism and to avoid pipeline delays. The register allocation and instruction scheduling phases actually impose constraints on each other, leading to conflicting requirements. It has been shown that integrating the two phases by introducing some form of communication between them leads to performance improvement. In the first part of the thesis, we evaluate four register allocation and instruction scheduling techniques in the context of *out-of-order* (o-o-o) issue processors. In the second part we discuss a very specific instruction scheduling technique known as *software pipelining* and special register renaming mechanism known as modulo variable expansion. In this part, we describe a new software pipelining approach which is sensitive to modulo variable expansion.

2. Evaluation of register allocation and instruction scheduling techniques

Several register allocation and instruction scheduling techniques have been proposed in the literature. Earlier works have studied the performance of these different techniques in the context of in-order issue superscalar or VLIW processors[1, 2]. However, since most modern processors (e.g. R10000, Alpha 21264) support o-o-o issue, an evaluation of different techniques in o-o-o issue processors is of considerable importance.

2.1. *Motivation*

In our study we have attempted to resolve two important issues. The first issue is regarding the complexity of the integrated compile-time techniques. An o-o-o issue processor does dynamic run-time scheduling with the help of run-time supports like register renaming and reorder buffer that remove the false dependences in the code. This, in spirit, is similar to the complex integrated techniques. Hence, it is interesting to see if complex compile-time techniques do improve the overall performance as considerably in o-o-o issue processor as in in-order issue processors. The second issue is whether in an o-o-o issue processor, instruction scheduling should be performed before or after the register allocation, i.e. whether a Pre- or Postpass-like approach should be adapted. The advantage of Postpass scheduling (i.e. when register allocation precedes scheduling) is that, though limited by the false dependences in the program, it does not introduce any additional spill code. The advantage of Prepass (i.e. when scheduling is done first) is that it can do more aggressive scheduling. However, such aggressive recording of instructions might cause spilling of registers. In an in-order issue processor, it is possible that the performance gain due to aggressive scheduling might be able to offset the performance degradation due to extra spills caused by the schedule. However, in an o-o-o processor, since scheduling is anyway done at run-time, it might do more harm than good to introduce spill code into the program. We address the above issues by comparing four techniques, namely, Prepass, Postpass, wherein register allocation precedes scheduling, Pinter's parallel interference graph technique which is a Postpass-like integrated technique and Integrated Prepass Scheduling, which is a Prepass-like integrated technique.

2.2. *Register allocation/Instruction scheduling methods*

In the Prepass strategy[3], instruction scheduling precedes register allocation. In the Postpass strategy, instruction scheduling is performed after the global register allocation phase. The register allocator and instruction scheduler used in both Pre- and Postpass methods are the same. The global register allocator is based on the technique described by George and Appel. The Postpass scheduler is essentially a list scheduler. The scheduler does basic block scheduling by constructing the data-dependence graph of the block, and maintaining a list of ready instructions it can schedule in each cycle. The integrated technique development by Pinter[4] is based on the coloring of the *parallel interference graph*. The graph provides a single framework within which the considerations of both register allocation and instruction scheduling can be applied simultaneously. An optimal coloring of this graph will ensure that no false dependence will be introduced. Readers are referred to Pinter[4] for further details. In Integrated Prepass Scheduling (IPS) instruction scheduling precedes register allocation; however, the scheduler is given a bound on the number of registers which guides it to increase parallelism when the register pressure is low and to limit the parallelism otherwise. Details of the method can be found in Goodman and Hsu.[3]

2.3. *Experimental framework*

We implemented the four techniques within the SUIF (Stanford University Intermediate Format) and Machine SUIF (MACHSUIF) compiler frameworks. SUIF accepts C or FORTRAN programs and generates equivalent code in the SUIF intermediate format. We used MACHSUIF, which provides the abstractions necessary to build machine-specific optimization passes. It consists of a code generator which accepts code in SUIF intermediate and translates it into an equivalent assembly language representation (OSF Alpha in our experimentation). In addition to the code generator and register allocator, MACHSUIF consists of the Control Flow Graph (CFG) library, the Data Flow Graph (DFA) library and a register allocator. We added two more modules, namely, the DDG Constructor and the List Scheduler to the existing modules in order to build integrated techniques. The techniques are applied on the output of the code generator, giving equivalent Alpha assembly programs of the benchmark. The assembly programs are then assembled on an Alpha 3.1 assembler to produce object code compatible with the SimpleScalar processor simulator. The SimpleScalar tool set provides five execution-driven processor simulators including a detailed out-of-order issue processor simulator. It supports out-of-order issue and execution by using an RUU (Register Update Unit) to automatically rename registers and hold the results of pending instructions. Our workload consisted of the first ten of the Livermore loops, all of which have large basic blocks and ten other benchmarks containing average or small-sized basic blocks. Some of the benchmarks are Queens, Bubblesort, SAXPY, SAXPY unrolled sixteen times Matmult, Primality, Hanoi and Fibonacci.

3. Results

We first evaluated the performance to the techniques with 32 integer and 32 floating-point registers made available to the register allocator. With this register availability, we expect the spill code to be minimal in our benchmarks. We found that in o-o-o issue processors all the methods area able to achieve the same performance in the Livermore loops. However, in the toy benchmarks, the performance of Prepass and IPS are slightly lower than Postpass. Next, we focus on benchmarks with a higher register pressure where the introduction of spill code may influence the performance. For this purpose, we used the same set of benchmarks used in the earlier section, but with reduced number of registers made available to the register allocator. We observe that Prepass and Integrated Prepass scheduling methods perform poorly compared to Postpass method. In particular, the performance of Prepass scheduling in o-o-o processors for Livermore loops is lower by 3% to 8%. In an in-order issue processor the performance degradation (compared to Postpass) is not that high (3% to 4.5%). Another observation that we make is that in an o-o-o processor the limitations of Postpass-like scheduling (limitation in instruction scheduling due to anti- and output-dependences) can be overcome with large RUU size, and greater issue widths. Next we compare the performance of Prepass schedules with codes in which no instruction scheduling was performed. We see that in an in-order processor, Prepass has better performance whereas in o-o-o- processor it is the other way around.

4. Modulo-variable expansion sensitive scheduling

The second part of the thesis deals with a very specific instruction-scheduling technique known as *software pipelining*.[5] Software pipelining or modulo scheduling is an aggressive instruction scheduling technique that exploits ILP in loops by overlapping the execution of successive iterations of loop. Due to the overlapping nature of software pipelining, it is possible that a variable is reassigned a new value even before the last use of the previous definition, i.e. the lifetime of variable can overlap with the subsequent definition of itself. Such lifetimes are called overlapping lifetimes.

Modulo-Variable Expansion (MVE) is a special register renaming technique that can be used to handle the overlapping lifetimes that occur in software pipelined loops. Traditional approaches to MVE unroll the constructed software pipelined *schedule* and rename the multiple definitions to avoid overlapping lifetimes. This approach limits the quality of the final unrolled schedule. We propose a new scheduling method which is sensitive to MVE, and hence called ‘Modulo-variable expansion sensitive software pipelining’. The proposed scheduling approach attempts to remove the limitations that currently exist with the traditional approaches to perform MVE. In the approach we unroll the *Data-Dependence Graph (DDG)* of the loop and reschedule it with the MVE-sensitive scheduler. The MVE-sensitive scheduler is based on Huff’s lifetime sensitive modulo scheduling algorithm.[6]

Our approach could lead to better schedules of the loop as compared to the traditional approach. In addition, our scheduling approach facilitates applying of several loop-unrolling-based optimizations which are not possible in the traditional approach. We implemented the technique and performance on 1450 benchmark loops and on three different machine models. Our results showed that a large number of the loops required MVE. Our scheduling technique improved performance only in a small number of the loops (6%), although the performance benefits in those loops were quite significant (12%).

5. Conclusions

In the first part of the work we evaluated four register allocation and instruction scheduling techniques in the context of o-o-o issue processors. We conclude that (i) integrated techniques lead to insignificant performance improvements in o-o-o issue processors. (ii) Postpass and Postpass-like methods perform better than Prepass and Prepass-like methods in o-o-o issue processors in the presence of high register pressure. (iii) Lastly, code produced by Prepass performed worse than code generated with **no** instruction scheduling in o-o-o issue processors.

In the second part of the work we proposed a new software pipelining technique which is sensitive to modulo-variable expansion. Our approach attempts to remove the limitations that currently exist with the traditional approaches to perform MVE. Although a large number of loops required MVE, our scheduling technique resulted in performance improvements only in a small number of them. However, the performance benefits in those loops were quite significant. In addition, our scheduling approach facilitates the application of several loop-unrolling-base optimizations in all loops that require MVE, which are not possible in the traditional approach.

We evaluated a few ILP compilation methods on o-o-o issue processors. However, in all the methods that we considered, instruction scheduling was limited to basic blocks. To achieve greater improvements in performance in o-o-o issue processors, scheduling techniques beyond basic blocks are important. Hence an evaluation of techniques that exploit ILP across basic blocks will be an interesting extension to our work. We observed that Postpass-like techniques lead to better performance than Prepass-like techniques. This is because the spill code introduced by the Prepass-like techniques more than offsets the performance gains due to aggressive scheduling. Hence, future techniques which reorder instructions first to remove spill code as much as possible will lead to greater performance. Lastly, our MVE-sensitive scheduling technique facilitates applying several unrolling-based optimizations. However, in our implementation we have not included any of them. As future work, these optimizations can be applied to the unrolled loop before MVE-sensitive scheduling. This will further improve the performance of MVE-sensitive software pipelining.

References

1. Bradlee, D. G., Eggers, S. J. and Henry, R. R. Integrating register allocation and instruction scheduling for RISCs, *Proc. Fourth Int. Conf. on Architectural Support for Programming Languages and Operating Systems,* Santa Clara, CA, Apr. 8–11, 1991, pp. 122–131.

2. Norris, C. and Pollock, L. L. An experimental study of several cooperative register allocation and instruction scheduling strategies, *Proc. 28th A. Int. Symp. on Microarchitecture,* Ann Arbor, MI, Nov. 29–Dec.1, 1995, pp. 169–179.

3. Goodman, J. R. and Hsu, W-. C.	Code scheduling and register allocation in large basic blocks, *Proc. 1988 Int. Conf. on Supercomputing*, St. Malo, France, July 4–8, 1988, pp. 442–452.
4. Pinter, S. S.	Register allocation with instruction scheduling: A new approach, *Proc. ACM SIGPLAN '93 Conf. on Programming Language Design and Implementation,* Albuquerque, NM, June 23–25, 1993, pp. 248–257.
5. Rau, B. R. and Fisher, J. A.	Instruction-level parallel processing: History, overview and perspective, *J. Supercomputing,* 1993, **7**, 9–50.
6. Huff, R. A.	Lifetime-sensitive modulo scheduling, *Proc. ACM SIGPLAN '93 Conf. on Programming Language Design and Implementation,* Albuquerque, NM, June 23–25, 1993, pp. 258–267.

Thesis Abstract (M. Sc. (Engng))

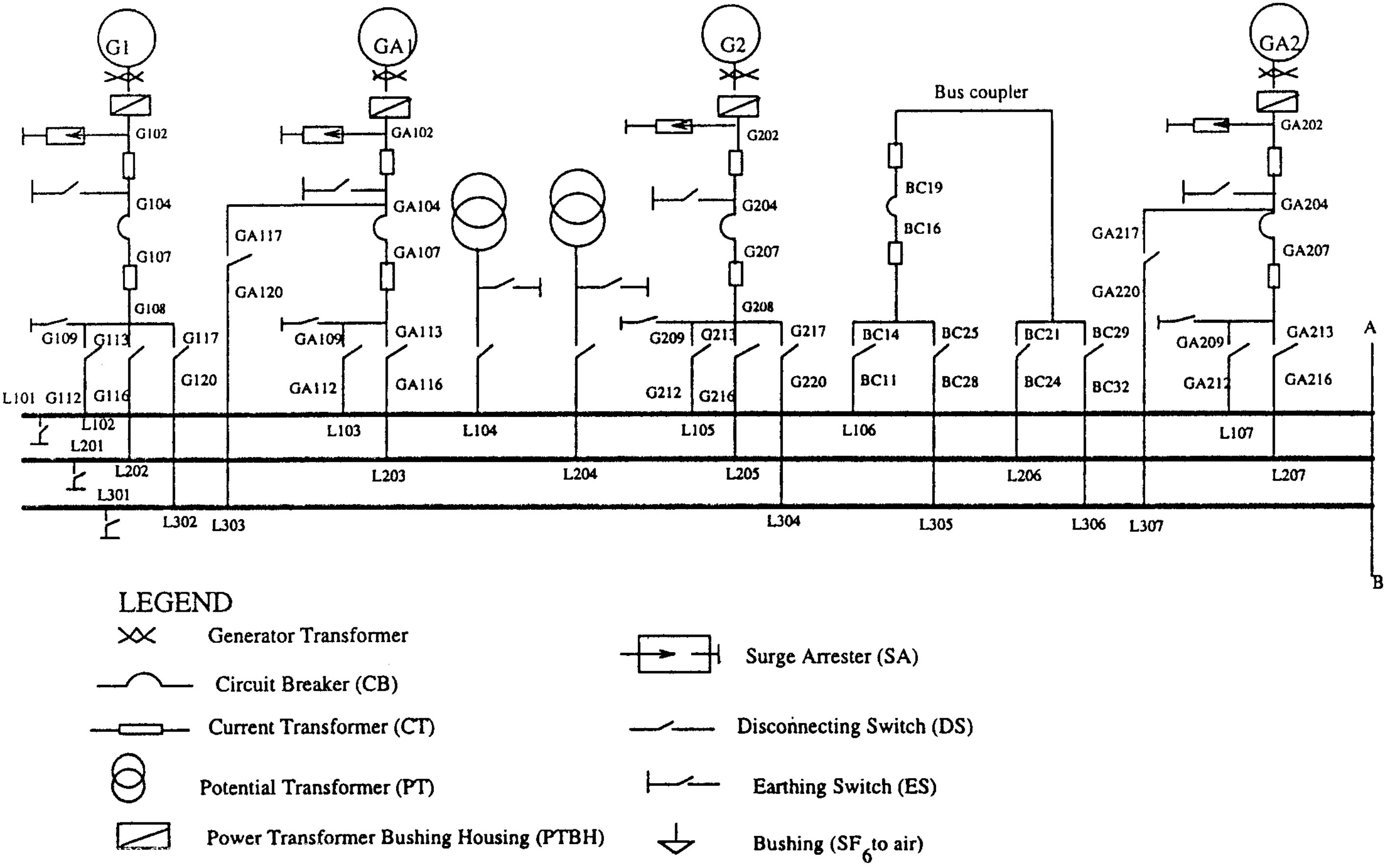
Estimation and measurement of very fast transient overvoltage (VFTO) in a gas insulated substation (GIS) by V. Vinod Kumar
Research supervisor: Dr M. Joy Thomas
Department: High Voltage Engineering

1. Introduction

A gas-insulated substation (GIS) consists of a grounded metallic chamber enclosing a high voltage bus with SF_6 gas at high pressure (4–6 bars) used as the insulating medium in between the two. GIS technology is gaining popularity due to several advantages and is replacing the conventional air-insulated substations (AIS). In spite of various advantages, GIS has many reliability-related problems, remedial measures for which are yet to be developed. Several failures have been reported in GIS rated for voltages above 240 kV when either circuit breaker or disconnector switching operation takes place. These failures have been attributed to very fast transient overvoltages (VFTO) which are generated during the switching operation. VFTO is characterized by very fast rise-time of the order of 10 ns or less and followed by high-frequency oscillations of several kHz to MHz. VFTO magnitude depends on the substation layout, the switching condition as well as the trapped change on the open end of the GIS bus. VFTO can lead to transient enclosure voltage (TEV) as well as electromagnetic compatibility (EMC)-related problems to sensitive electronic circuits used in metering and controlling equipment. Hence, it is important that an estimate of the VFTO magnitude is made before commissioning of substations. Also, while commissioning, it becomes necessary to measure the VFTO magnitudes and compare them with the estimated values. The work reported here focuses on the computation of VFTO magnitudes for a 420 kV GIS as well as development of a sensor for measurement of VFTO.

2. Computer simulation

A one-line diagram of the chosen 420 kV GIS is shown in Fig. 1. This substation has four main generators (G1 to G4) and four auxiliary generators (GA1 to GA4) feeding the three buses and three outgoing feeder lines from the substation. Computer simulations have been using EMTP and appropriate transient models for the GIS components.[1,2] A total of 61 different switching conditions have been simulated for different trapped charge magnitudes of 0.3, 1.0 and 1.2 p.u.


G1
GA1
G2
GA2
Bus coupler
G102
G104
G107
G108
G109
G113
G117
G112
G116
G120
GA102
GA104
GA107
GA109
GA112
GA113
GA116
GA117
GA120
G202
G204
G207
G208
G209
G212
G213
G216
G217
G220
BC19
BC16
BC14
BC11
BC25
BC28
BC21
BC24
BC29
BC32
GA202
GA204
GA207
GA209
GA212
GA213
GA216
GA217
GA220
A
B
L101
L102
L201
L202
L301
L302
L303
L103
L104
L105
L106
L107
L203
L204
L205
L206
L207
L304
L305
L306
L307
LEGEND
Generator Transformer
Circuit Breaker (CB)
Current Transformer (CT)
Potential Transformer (PT)
Power Transformer Bushing Housing (PTBH)
Surge Arrester (SA)
Disconnecting Switch (DS)
Earthing Switch (ES)
Bushing (SF$_6$to air)

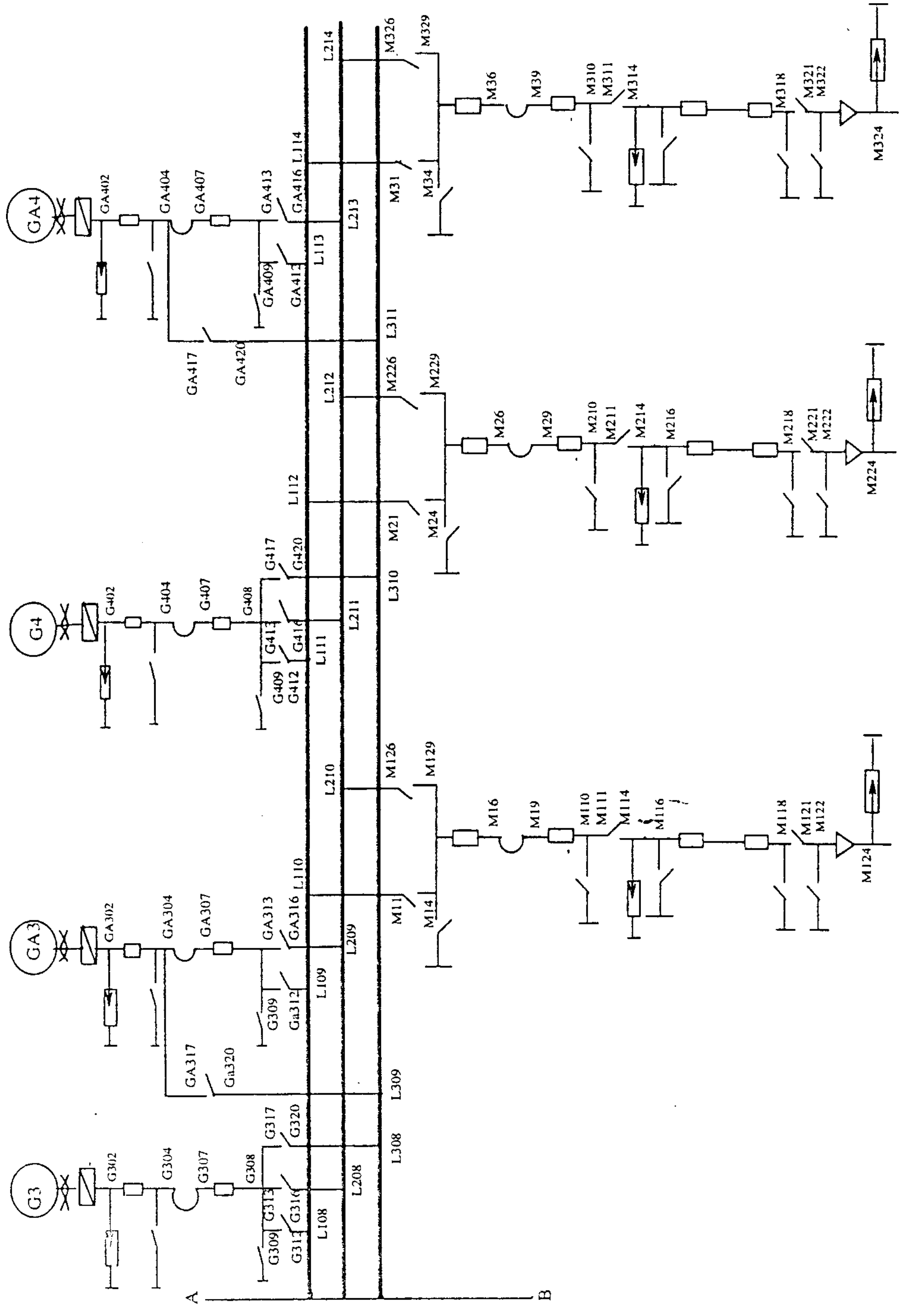


FIG. 1. Single line diagram of the 420 kV substation.

3. Results and discussion

Only two specific cases have been presented and discussed here. The first is, all generators connected and the disconnector switch between M226 and M229 operated with circuit breaker between M26 and M29 are kept open with a trapped charge of 0.3 p.u. and the second is, all generators connected to the bus and the circuit breaker between M26 and M29 operated with a trapped charge of 0.3 p.u.

3.1. *VFTO waveforms*

The computed VFTO waveforms for disconnector and circuit-breaker switching conditions are shown in Fig. 2. In general, it is seen that when the observation node is close to the switching point, VFTO waveform has higher frequency content as compared to a farther away node from the switching point. Also, it is found that the VFTO peak magnitudes are higher in the case of circuit-breaker operation as compared to that of disconnector switching operation. Out of all the 61 switching operations simulated, amongst disconnector switching operations, VFTO peak is found to be the highest when the disconnector switch between G320 and G317 is operated with the circuit breaker kept open between G307 and G304 and all the generators except G3 are energised with an assumed trapped charge of 1.2 p.u. Its value is 2.25 p.u. (node G305). In the case of circuit-breaker operations, the VFTO peak magnitude is found to be maximum for circuit breaker operated between GA204 and G207 and only GA2 line is connected and energized with a trapped charge of 1.2 p.u. Its value is 2.87 p.u. at the node M324 on the outgoing feeder $X=3$.

3.2. *Variation of VFTO peak along the nodes*

The variation of the VFTO peak along the nodes of the bus for each of the above-mentioned switching conditions is shown in Figs 3 and 4. There is a distinct pattern of variation of VFTO peak along the nodes of the GIS in the case of disconnector switch operation as compared to that of circuit-breaker operation (compare Figs 3 and 4). From Fig. 3 it is seen that the main generator line G4 ($X=4$) shows a maximum VFTO level of 1.27 p.u., the auxiliary generator line GA4 ($X=4$) slightly more than 1.21 p.u. and the outgoing feeder line M2 close to 1.5 p.u. The VFTO magnitudes reduce near the open circuit breaker and once again increase near the junction of the overhead line with the GIS (at node M224).

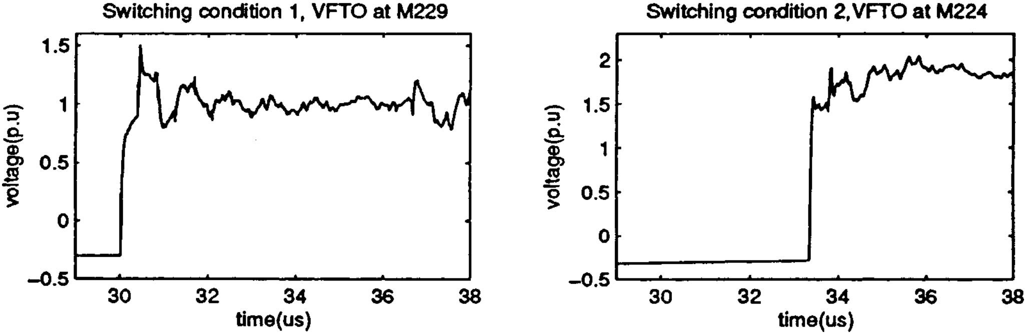


FIG. 2. VFTO waveforms for a trapped change of 0.3 p.u.

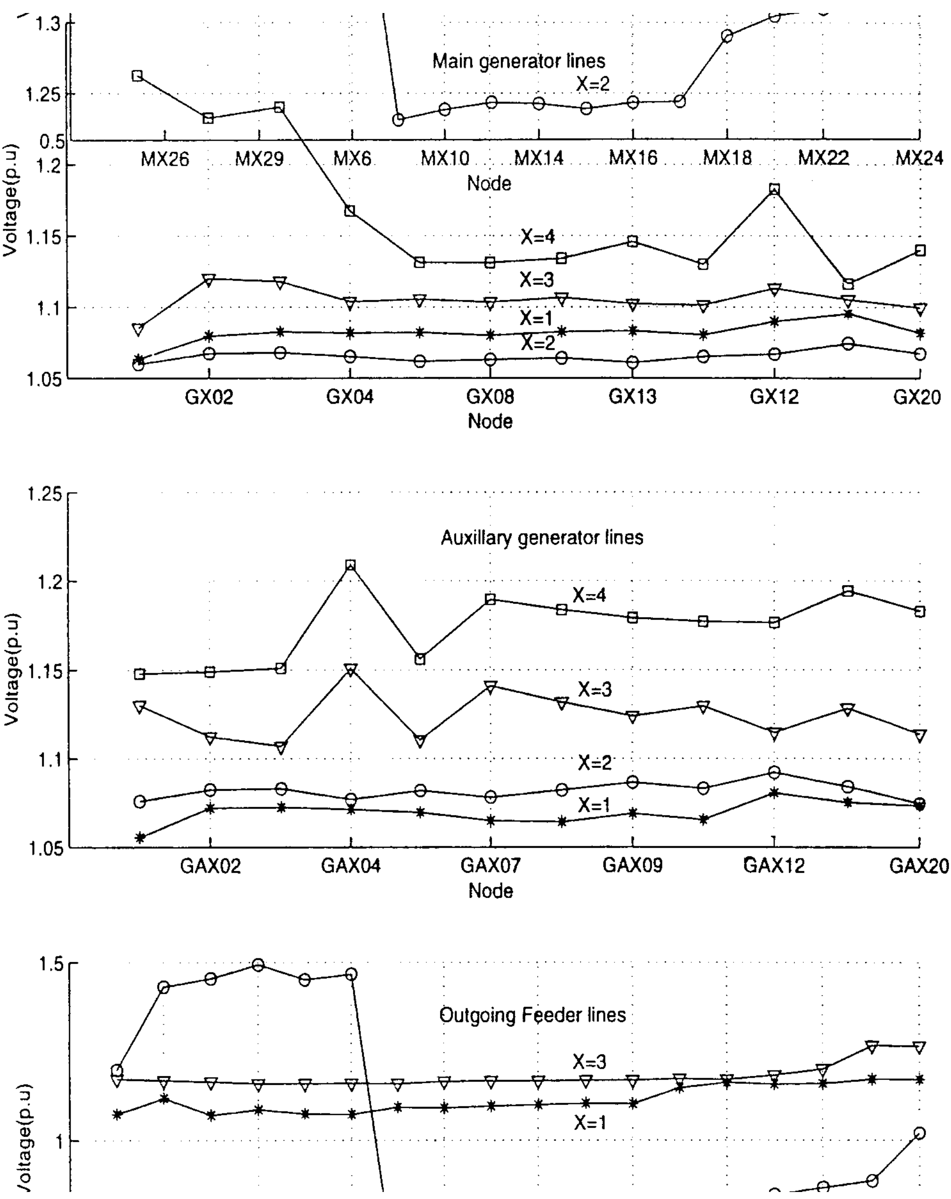


FIG. 3. Variation of VFTO peak along the nodes for switching case no. 1 with a trapped charg of 0.3 p.u. (disconnector operation).

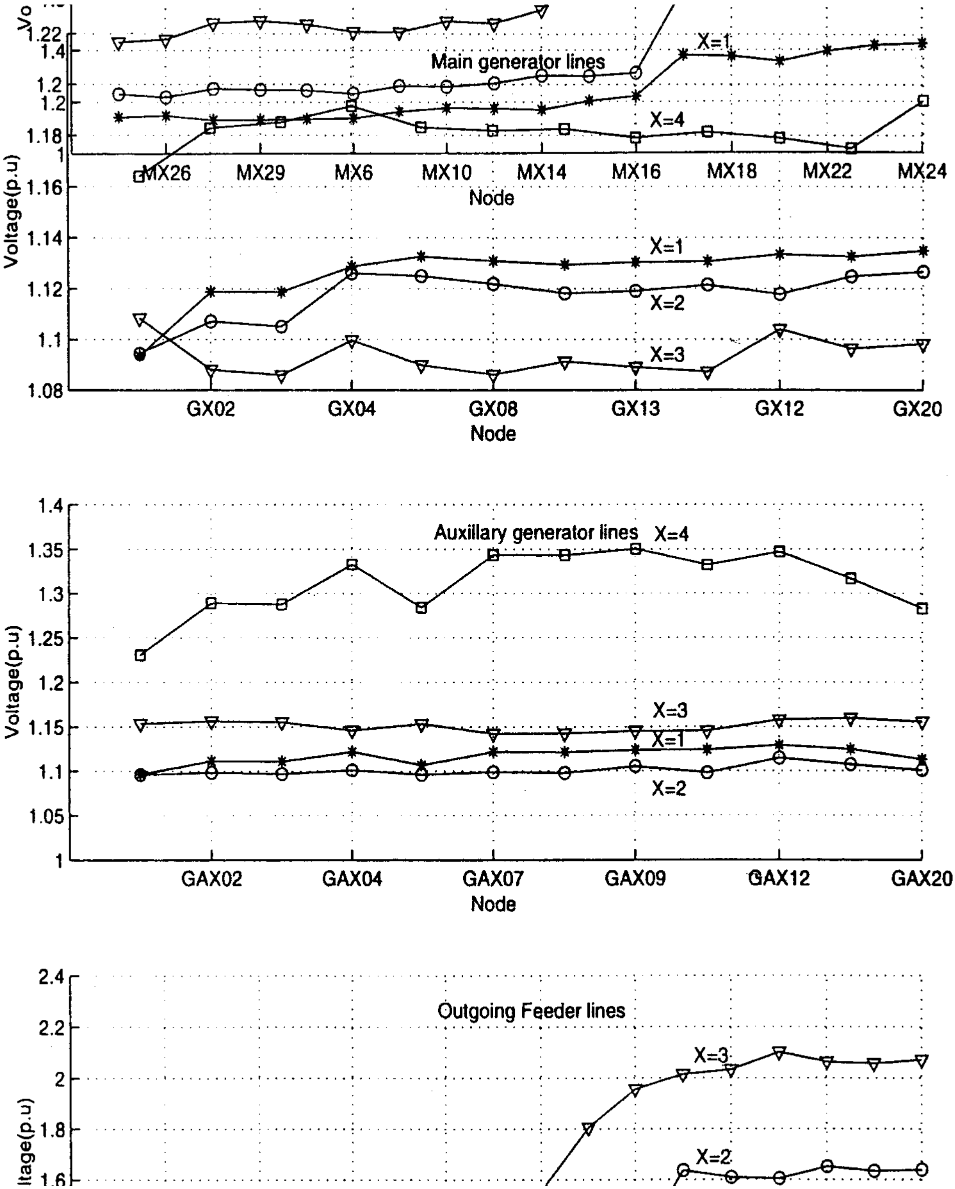


FIG. 4. Variation of VFTO peak along the nodes for switching case no. 2 with a trapped charge of 0.3 p.u. (circuit breaker operation).

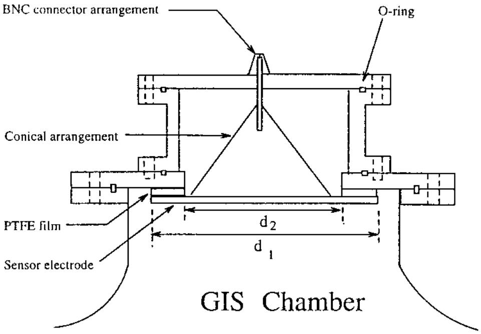


FIG. 5. Sketch showing the capacitive sensor.

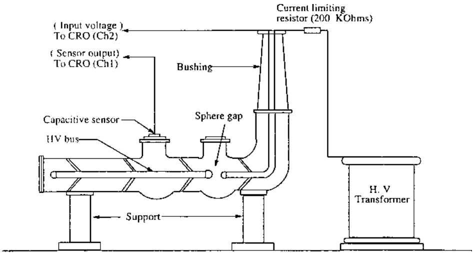

Current limiting resistor (200 KOhms)
(Input voltage) To CRO (Ch2)
(Sensor output) To CRO (Ch1)
Bushing
Capacitive sensor
Sphere gap
HV bus
H. V Transformer
Support

FIG. 6. Experimental arrangement of the GIS test set-up.

From Fig. 4, it is seen that the main generator line G4 (X=4) and auxiliary generator line GA4 (X=4) have maximum VFTO levels of 1.19 and 1.3 p.u, respectively, whereas the outgoing feeder line M3 (X=3) has a value of 2.1 p.u. There is a steady increase in the VFTO levels beyond the switching point till the junction of GIS with overhead line in the case of circuit-breaker switching operation.

3.3. *Effect of trapped charge on the VFTO levels*

The variation of difference in VFTO peak along the nodes was calculated with change in trapped charge from 0.3 p.u. to 1.2 p.u. It is noted that there is maximum change in peak VFTO level on the line where the switching operation is done and the change is more for line where the VFTO levels are higher. The change in peak VFTO levels is not found to be proportional to the change in trapped charge values.

4. Development of a capacitive sensor for simultaneous measurement of 50 Hz restrike pattern as well as VFTO in a GIS

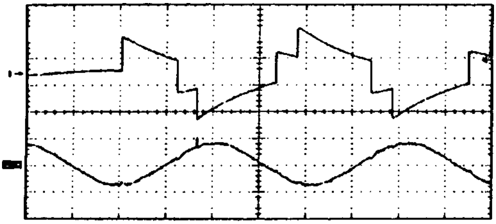
A capacitive sensor based on the theory of potential measurement using floating electrode has been developed. The high-voltage capacitor (C_1) of the capacitive divider is the capacitance between the high-voltage bus and the sensor electrode with SF_6 acting as the dielectric in between them.

A low-voltage capacitor (C_2) is formed between the sensor electrode and the ground. PTFE (Polytetrafluoroethylene) film is used as the dielectric between the sensing electrode and the ground (see Fig. 5). The output is measured across the low-voltage capacitor through a conical termination (Fig. 6) to avoid impedance mismatch.

Calibration of the sensor has been done with sinusoidal, impulse and step voltages with the capacitive sensor mounted on a 145 kV-rated GIS test set-up and the ratio of the sensor is found to be 456. A sketch of the GIS test set-up is shown in Fig. 6.

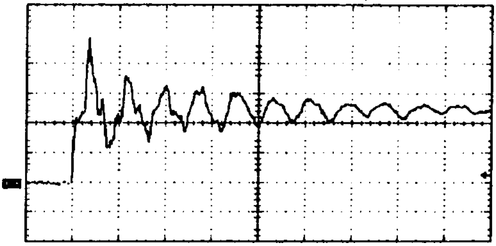
4.1. *Measurement of 50 Hz restrike pattern and VFTO in the GIS test set-up*

The HV bus was excited with a sinusoidal voltage of 50 Hz from a 150 kV, 50 kVA testing transformer. The sphere gap shown in Fig. 6 is used to simulate the disconnector switching



Ch1: 20V/div (sensor output), Ch2: 25 kV/div (HV bus voltage), time: 5 ms/div.

FIG. 7. Oscillagram of restrike pattern from the sensor.



Ch1: 10V/div (sensor output), time: 50 ms/div.

FIG. 8. Oscillagram of VFTO measured by the sensor.

operation and thereby the generation of VFTO. The VFTO patterns generated were recorded using the sensor developed (Figs 7 and 8). It can be seen from Fig. 7 that there is always more than one restrike which ultimately leads to the step-shaped waveform or the restrike pattern. Every time there is a restrike, a VFTO is generated as shown in Fig. 8, which was also successfully captured using the same capacitive sensor developed.

5. Conclusions

The VFTO peak magnitudes are higher close to the switching point in the case of disconnector operations, whereas they are higher at the junction of the GIS with overhead line in the case of circuit-breaker operation. Also, close to the switching point, VFTO waveform has higher frequency of oscillations as compared to a farther away observation point. There is also a distinct pattern of variation of VFTO peak along the nodes of the GIS in the case of disconnector switch operation as compared to that of circuit breaker operation. The circuit breaker operation results in the highest VFTO level amongst all the switching operations done. When the trapped change on the line is increased, there is an increase in the VFTO levels, but the increase is not proportional to the magnitude of the trapped charge.

The capacitive sensor developed is wide band sensor which can measure both the low frequency restrike pattern as well as the ns rise-time VFTO. Also the divider ratio of the sensor is found to be constant at 456±2 for a wide range of frequencies.

References

1. LEWIS, J., PRYOR, B. M., JONES, C. J. AND IRWIN, T. Disconnector operations in gas insulated substations overvoltage studies and tests associated with a 420 kV installation, *CIGRE*, Vol. II, 1988, paper 33.09, pp. 1–8.

2. *Insulation co-ordination of GIS: Return of experience on site tests and diagnostic techniques*, Joint Working Group 33/23.12, Electra, February 1998, pp. 67–97.

Thesis Abstract (M. Sc. (Engng))

Design support system to integrate computer-aided modeling and analysis by G. Ranganath
Research supervisor: Prof. B. Gurumoorthy and Dr M. S. Rajagopal (KEC)
Department: Mechanical Engineering

1. Introduction

In an engineering industry, to reach the market faster with better quality products, companies rely heavily on automation of design. To succeed in today's competitive world market, product developers are under tremendous pressure to shorten the product development cycle and to improve product quality. While CAD tools have contributed to raising the effectiveness and productivity of designers, it is being increasingly realized that extensive training is required for the user to exploit the features in these systems to increase productivity and these tools require considerable customization for them to be used effectively in each industrial segment. With increasing use of these tools, another problem has come to light, i.e. transition from one task to another. There is thus a need for a support system to enable the designer use these computer tools more effectively. The focus of this work is on the development of such a design support system, which improves the design artifact by enhancing the design process. To overcome the deficiencies and increase the productivity of the designer–design parameterization, associativity of product data maintenance between different tasks, and capturing rules of thumb/heuristics used in the design-techniques are investigated. This addresses the issue of effective use of the computer-based tools and ease transition between tasks.

2. Literature review

The earlier efforts can be classified into the following categories:

Case-based approach: Here solution to earlier case is retrieved from memory based on similar to the current case to solve a new design problem.[1]
Agent/object-based approach: In this, design knowledge is treated as an object, which is mainly applicable to process industries. Here, design knowledge is treated as an object and it is a natural extension to the paradigm of chemical agents, which means conversion of profitable new reactions discovered by chemists into process flow sheets.[2]
Concurrent engineering approach: In this type of approach design modifications are concurrently examined for suitability with respect to some other tasks (mostly manufacturing). The basic idea in this approach is to automate or provide computer-based support to enhance automation in transition between tasks in product development.[3]

It is found that most of the work to date on design support system has been limited to database management or design model in continuous process industries. Hence there is a need to develop a design support system for engineering industries involved in the manufacture of discrete goods.

3. Design support system (DSS)

CAD tools, such as computer-aided drafting, solid modeling, and finite-element programs, currently used are independent and to use these tools effectively in product design, users have to be well trained in each of these tools. To integrate the above-mentioned isolated CAD tools effectively, a design support system has been developed (Fig. 1).

The DSS aims at reducing the user intervention required in the tasks of data preparation, data presentation and interpretation. Figure 2 describes the DSS scheme.

The first main module is the model assembler (MA). In this module, the user is assisted in constructing the CAD model of the product in a modular manner that supports description of the

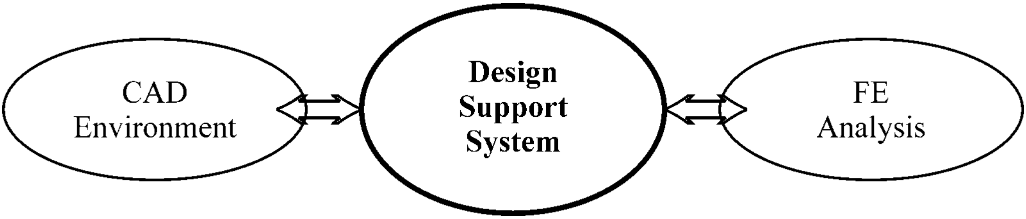


FIG. 1. Design Support System.

product at different levels of detail. The model assembler also ensures parametric construction so that design can be scaled up or down automatically based on functional specification.

The second module is the *Result extractor*, which maps the results of the analysis as obtained from commercial systems in use today, on to the product/CAD model. Given a direct link between stress levels and product geometry (features) it is possible to link with some domain knowledge base to do automatic modification.

In the current work, the focus of the DSS has been on the following issues :

1. *Data preparation*: Here user is enabled to construct a model of the product that will allow realizing simplified model for analysis and adding information relevant to analysis such as

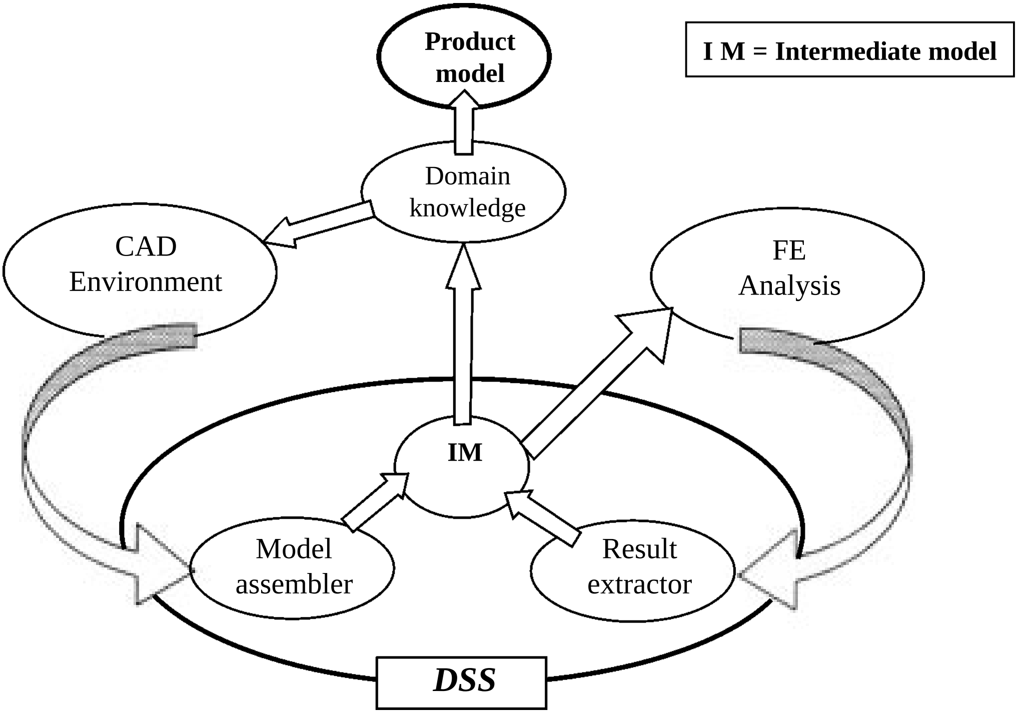


FIG. 2. Design support system scheme.

loads and boundary conditions. This is achieved by allowing part to be built modularly at a different level of detail by using layer and feature definition techniques, and providing means to define attribute handler in the CAD model. To simplify the part or assembly's geometry, suppression techniques are used to get the finite-element model.

2. *Data interpretation*: To interpret the FEA post-processing data, the algorithm is used in this DSS which gives the output as failure points.

3. *Data presentation*: In the current work, the proposed presentation method is superimposing the FEA results on CAD model. This gives the coordinates the critical points. By this, the feature or part in the product model can be accessed directly. This helps to modify the CAD model features in a better way.

3.1. *Implementation*

The domain chosen for implementation of the design support system is the design of a transformer tank structure. The parametric solid model of a transformer structure has been developed by using layer and feature definition techniques, using Pro-Engineer (Release18.0) in the IBM RS6000 workstation. Using suppression techniques, the CAD model is simplified to get a finite element (FE) model. Here four levels of suppression were required to get an FE model from the solid model. Figure 3 represents the first-level suppressed solid model of the distribution transformer. Final solid model has been transferred using IGES transfer to the analysis package. For analysis ANSYS 5.4 was made use of on IBM RS6000 workstation. After carrying out the static structural analysis, postprocessing results such as principal stresses are written on to a neutral file. DSS interprets the postprocessed data and output the coordinate points where the stresses have crossed the safe yield values on to a neutral file. These regions are highlighted on the solid model by reading the same in the solid modeler. This guides the designer to take suitable decisions to modify the base structure.

4. Contributions

The main contributions of the thesis are :

- Guided data preparation/definition of the CAD model: DSS provides a framework for the

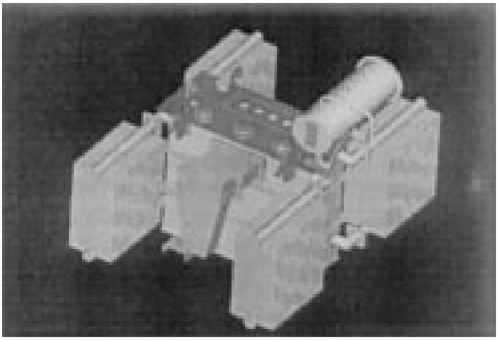


FIG. 3. First-level suppressed solid model.

user to construct CAD models that are amenable for analysis by supporting a layered feature-based construction so that models of increasing complexity can be made available for analysis.

- Algorithm to interpret the FEA results in a simplified manner, so that it can be incorporated with heuristics rules for design modification.
- Interface FEA results to CAD model and data presentation scheme on the CAD model for better interpretation of FE results and subsequent redesign/design modification.

The above are achieved by using commercially available computer-based tools.

5. Scope for future work

- Property data transfer using STEP standards.
- Defining the imaginary planes/surfaces where the region is critical on the product model using point/surface handlers, to prompt the user to control the resolution of the mesh (adaptive meshing) being generated for analysis.
- Knowledge-based systems to exploit a direct linkage between FE results and CAD geometry to suggest redesign/design modification.

References

1. Nakatani, Y., Tsukiyama, M. and Fukuda, T. Case organization in a case-based engineering design support system, *Conf. Proc. 1991 IEEE Int. Conf. on Systems, Man, and Cybernetics,* Vol. 3, pp. 1789–1794.

2. Han, C., Douglas, J. M. and Stephanopoulos, G. Agent based approach to a design support system for the synthesis of continuous chemical process, *Computers Chem. Engng,* 1995, **19**, S63–S69.

3. Roy, U. An intelligent interface between symbolic and numeric analysis tools required for the development of an integrated CAD system, *Computers Ind. Engng,* 1996, **30**, 13–26.

Thesis Abstract (M. Sc. (Engng))

A hybrid approach for automatic construction of solid models from two dimensional orthographic views by D. Mukherjee
Research supervisor: Prof. B. Gurumoorthy
Department: Mechanical Engineering

1. Introduction

Engineering drawing is an aid to describe objects and provides all possible information about the shape, size and process of manufacture of the object. Every solid object is represented by a set of projection drawings. With the rapid growth of computing power and the increasing trend towards automation in design and manufacture, the use of solid model representation in design, analysis, process planning and several other areas like machine vision is being preferred. There is a need therefore to convert the legacy data in the form of engineering drawings to 3D solid models. There are two different aspects to this problems. One involves the processing of the scanned drawings to separate the geometric entities from the annotational text and the other involves reasoning with vector output from scanning and image processing and constructing the solid model. The focus of this is on the geometry part of the problems only.

2. Literature review

The earlier efforts can be classified into the following categories.

- *Bottom-up approach*: Here 3D objects are reconstructed through the steps of obtaining 3D vertices, 3D wireframe, virtual faces, virtual blocks and final solid.[1,2]
- *Volume-based approach*: This approach relies on geometric sweep and boolean operations and later on compares the generated volume with an existing library of primitives (*variant*)[3] or generates the primitives based on loop information (*generative*).[4,5]

The present approach is a hybrid of the earlier two approaches and uses them selectively to exploit the advantages of both.

3. View segmentation and folding

All previous work in this area assumes input to be given in the form of a 2D wireframe where the coordinates of the vertices in the views are the coordinates of those vertices when placed in their actual view planes. View segmentation and folding locate the input views given with respect to a single coordinate frame (which is easier for the user to provide) in their actual view planes.

In view of segmentation we try to obtain an uniform spacing among the different views. A hypothetical line known as the common coordinate line is obtained between each pair of adjacent views and level-order traversal over the graph of view nodes is used to position the views to be equally spaced about the common coordinate line. The segmented views are then folded about the common coordinate lines in certain convention to obtain folded set of view planes (Fig. 1).

4. Hybrid algorithm for regular views

Our approach uses a combination of the concept of bottom-up and volume-oriented approaches. The idea is to apply the volume-oriented approach to as many cases of simple sweep and intersection (between matching pair of cycles with one cycle having a rectangular span) as possible to obtain swept elemental solids, the reason being the most practical objects contain considerable number of swept features. The swept elemental solids are then projected over the existing views and unaccounted edges in the input views are identified. Bottom-up procedure is applied over the unaccounted edges and 2D vertices to obtain bottom-up elemental solids. Finally, the swept and bottom-up elemental solids are combined using 3D boolean operations to obtain the final solid.

Repeated comparison of the projection of the generated solid and the input ensures correctness, whereas applying bottom-up for cases, where volume-based approach fails, guarantees completeness. Applying checks for sweep and volume-based concepts earlier, substantially reduces the problem size which has to be input for the bottom-up and thus speeds up the entire process.

5. Hybrid algorithm applied over auxiliary views

We propose extension of the algorithm proposed for regular views to cases with auxiliary views where views can be incomplete and on nonconventional view planes. Modification is necessary

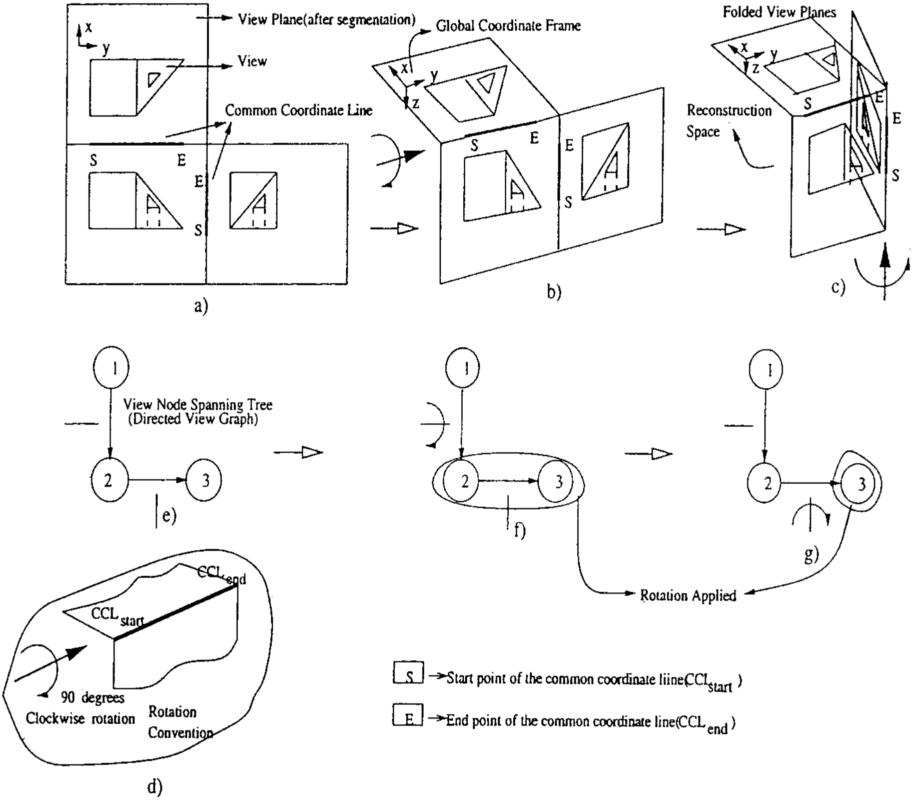


FIG. 1. Steps in view plane folding.

because of the incomplete nature of the auxiliary views. Here we apply a heuristic rule to separate out relevant part of the complete views which correspond to the incomplete views. Each of the incomplete views with its trimmed relevant part-views act as source of a separate part solid which is a part of the final solid. An object can have several such part solids. Part solids are projected over all complete views to identify input corresponding to the remaining part of the final solid. All part solids are finally added to form the final solid.

6. Results and discussion

The proposed algorithm was implemented on Sun UltraSparc workstation using Sparc CC Version 4.01 compiler in C programming language. The Geometric Modeling Toolkit SHAPES was used for geomteric operations like boolean operations and for display of the results generated. Following are the results of the implementation of our hybrid algorithm over a regular (Fig. 2) and auxiliary (Fig. 3) view problem.

6.1. *Contributions*

- An automatic procedure for segmenting the input views and consistently folding the viewplanes has been developed.

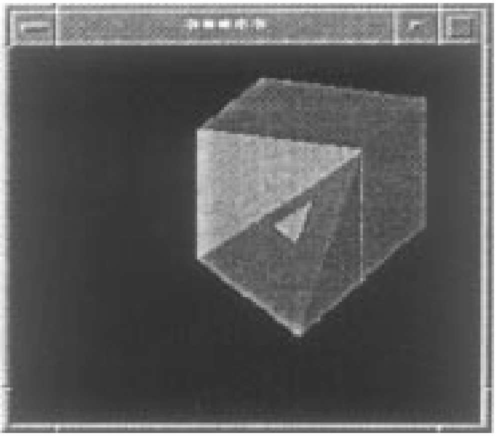


FIG. 2. Solid model obtained from a set of regular views.

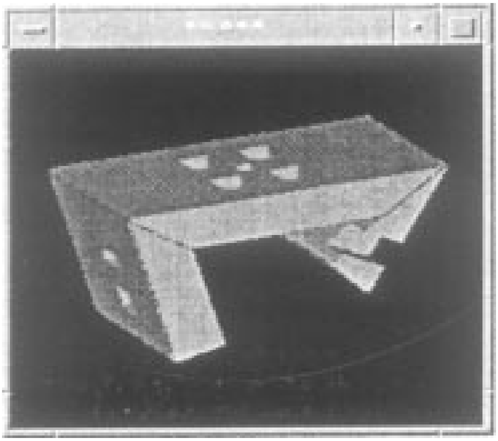


FIG. 3. Solid model obtained from a set of auxiliary views.

- Input is no longer expected from several local coordinate frames with 3D form of coordinates; instead it comes from a single global 2D frame of reference.
- A hybrid procedure to exploit the advantages of volume-oriented and classical bottom-up approach for constructing solid model from orthographic views.
- First effort to handle local and auxiliary views.

6.2. *Scope for future work*

- Automatic identification of completed and incomplete views.
- Identifying sweeps other than extrusion (rotational sweep or sweep with varying cross-section) and determining parameters required to describe them.
- Use of NURB (nonuniform rational B-splines) or SHGC (straight homogeneous generalized cylinder) to handle curved and freeform solids.
- Interfacing between a drafting system and the programming environment.
- Addressing the uniqueness problem of a solid and its projection.

References

1. Idesawa, M. — A system to generate a solid figure from three views, *Bull. Jap. Soc. Mech. Engrs*, 1973, **16**, 216–225.

2. Wesley, M. A. and Markowsky, G. — Fleshing out projections, *IBM J. Res. Dev.*, 1981, **25**, 934–954.

3. Aldefeld, B. — On automatic recognition of 3D structures from 2D representations, *Computer Aided Des.*, 1983, **15**, 371–380.

4. Saha, K. and Gurumoorthy, B. — Solid model construction from 2D information, *ASME Conf. on Design Automation*, 1994.

5. Manoj — *Automatic construction of solid models from 2-dimensional orthographic projections*, M.Sc. (Engng) Thesis, Department of Mechanical Engineering. Indian Institute of Science, Bangalore, 1997.

Thesis Abstract (M. Sc. (Engng))

Parametric blending of surfaces by R. V. Prasad
Research supervisor: Prof. B. Gurumoorthy
Department: Mechanical Engineering

1. Introduction

Blending is a term used mainly in the field of computer-aided geometric design for the creation of localized (smooth) transitions between surfaces. Blends are intermediate surfaces which smoothly join other surfaces of the exterior of an object.[1] Often, some of the original surfaces of the object would have intersected in sharp edges, and, for a variety of reasons, technical, aesthetic or otherwise, it may be desired to replace some or all of the sharp edges by corresponding smooth faces or blends. These surfaces meet the rest of the object with (at least) G^1 continuity. In general, it may not be preferred to just smooth separate edges, but also vertices where they meet, or even a whole region of the original surface of the object. Edge blending is becoming a necessary operation in geometric and solid modeling in which sharp edges must often be rounded or replaced by a smooth surface for aesthetic or functional reasons. Edge blending has been carried out using many standard techniques, but the blending of vertices where many edges meet has not been addressed very effectively. This work aims at developing a technique for vertex blending. Since blending operation involves creation of composite parametric patches, ensuring continuity across each surface becomes very important. Tangential continuity across patches may not be enough for some applications here, hence higher order continuities across patches are desirable.

2. Literature review

Different technique have been proposed for vertex blending. Conventional vertex blending techniques have many limitations as nonorthogonal edges. Adjacent edges blends having a large difference in their radii yield vertex blends which are not aesthetically pleasing and concave–convex edge configurations are not handled well. Setback vertex blending technique can handle all the above limitations. The technique proposed[2] makes use of dummy edges for handling concave angle configurations. The proposed technique of setback vertex blending overcomes the above-mentioned limitations of the previous schemes and proposes a technique wherein minimum number of patches are created at the vertex. This technique makes use of a triangular Bézier patches.

Previous techniques proposed for ensuring continuity between adjacent patches has been limited to either Bézier patches[3] or providing a validation check whether two adjacent parametric (Bésier[4] or B-spline[5] patches) patches are C^2 continuous. The technique proposed provides two constructive schemes for ensuring C^2 continuity between adjacent patches.

3. Setback vertex blending

In the proposed technique, the creation of spring, profile and interior boundary (IB) curves is done by using the same procedure that Varady and Rockwood have proposed. But a different procedure to create the vertex blend with minimum number of patches is attempted. This is achieved by the use of triangular Bézier patches along with four-sided patches for blending the vertex region. Due to the introduction of IB curves the vertex region is an *n*-sided polygon. It is possible to obtain the boundary of four- or three-sided patches from this polygon. The algorithm automatically chooses the number of triangular and rectangular patches to be created at the vertex, given the number of edges meeting at the vertex. If each IB curve at the vertex blend is assumed as the side of polygon, minimum number of four- and three-sided regions required to fill the polygon is created. Triangular Bézier patches are constructed only for vertices where odd number of edges meet. Instead of the dummy edge introduced by Varady and Rockwood[2] when concave configurations are encountered, the proposed technique handles them in a normal way. No dummy edges or zero width is introduced.

Creation of setback vertex blends involves the following steps:

1. Computer profile curves.
2. Computer spring curves.
3. Set fullness parameters and compute surface curvature at the base point.
4. Create interior boundary curves.
5. Create the control polygon for the central vertex blend.
6. Create setback patch ensuring G^1 continuity with respect to vertex patch.

Steps 1 to 4 are similar to the one proposed by Varady and Rockwood,[2] the only difference being that all the half spring, profile and interior boundary curves constructed will be of degree three (cubic).

4. Ensuring C^2 continuity between parametric surface patches

Two methods for ensuring C^2 continuity across adjacent B-spline patches are proposed.

4.1. *Using derivatives*

This method uses the technique of equating the derivatives for the two patches at the two ends of the common boundary curve.

4.2. *Using twist vector*

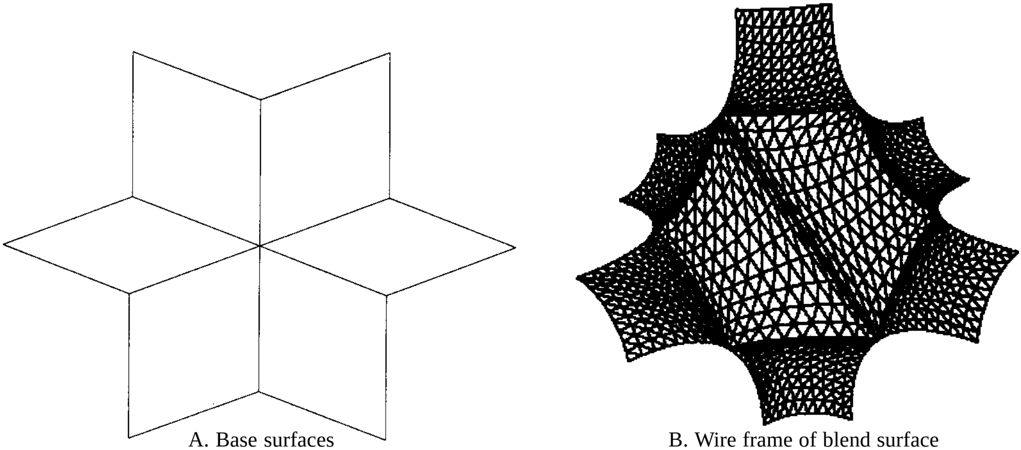
In the second method, the geometric interpretation of twist vector is made use of for ensuring C^2 continuity.

5. Results and discussion

The algorithms for setback vertex blending and ensuring C^2 continuity between patches were implemented using C programming language on an Indigo 2 machine. For display, Open Inventor and its graphics library were made use of. The results of the setback vertex blending implementation for a concave–convex configuration are shown in Fig. 1. The two constructive procedures for ensuring C^2 continuity between patches have been implemented.

5.1. *Contributions*

- A technique has been proposed for setback vertex blending where triangular patches are used. Compared to the previous technique the proposed scheme makes use of lesser number of patches to get the vertex blend.



A. Base surfaces

B. Wire frame of blend surface

FIG. 1. Setback vertex blending for 6-edge concave–convex configuration.

- Odd-numbered edge configurations can be handled much easily than the previously proposed technique and need not create any dummy edges. First-order continuity between the patches is ensured.
- The module to convert the triangular Bézier patch to three rectangular Bézier patches has been implemented, so that the proposed scheme can be used on any existing CAD modeller.
- Two constructive procedures are proposed for ensuring curvature continuity between two adjacent B-spline patches.
- This is the first constructive scheme for ensuring curvature (C^2) continuity.

5.2. *Scope for future work*

- Further work on setback blending can be focussed on ensuring C^2 continuity with triangular Bézier and B-spline surfaces.
- Shape optimization can be another field which needs to be looked into.
- Ensuring C^2 continuity between adjacent patches can be extended to higher order of continuities between patches.
- The proposed technique considers one patch as the reference, and modifies the other patch for ensuring curvature continuity. The technique can be used for obtaining the continuity by modifying both the patches, such that an optimized change takes place in both the patches.

References

1. VIDA, J., MARTIN, R. R. AND VARADY, T. A survey of blending methods that use parametric surfaces, *Computer Aided Des.,* 1994, **26**, 341–365.

2. VARADY, T. AND ROCKWOOD, A. Geometric construction for setback vertex blending, *Computer Aided Des.,* 1997, **29**, 413–425.

3. DEROSE, T. D. Necessary and sufficient conditions for tangent plane continuity of Bézier surfaces, *Computer Aided Geometric Des.,* 1990, **7**, 165–179.

4. Pegna, J. and Wolter, F.	Geometrical criteria to guarantee curvature continuity of blend surfaces, *ASME Trans. J. Mech. Des.*, 1992, **114**, 201–210.
5. Ye, X.	The Gaussian and mean curvature criteria for curvature continuity between surfaces, *Computer Aided Geometric Des.*, 1996, **13**, 549–567.

Thesis Abstract (Ph. D.)

Propagation of a curved weak shock by Monica Anand
Research supervisor: Prof. Phoolan Prasad
Department: Mathematics

1. Introduction

Formation of a kink and its propagation on a shock front were experimentally studied by Sturtevant and Kulkarni.[1] For a weak shock (but not so weak that it follows the linear theory of acoustics), rays of the converging shock front curve away from the focal region and the shock front emerges flattened with no loop, so that caustic is resolved. In this case, a smooth concave shock front with continuous amplitude distribution on it develops a pair of kinks after some time. Across the kink, the shock strength and the direction of normal to the shock change discontinuously. Kinks appear not only on a shock front but also on a one parameter family of weakly nonlinear wavefronts in high frequency approximation as studied by Prasad and Sangeeta.[2]

2. Governing equations

A system of shock ray equations consists of ray equations derived from a shock manifold partial differential equation (or eikonal equation for a shock front) and an infinite system of compatibility conditions along a shock ray. These compatibility conditions are derived from the equations governing the motion of the medium in which the shock propagates.

We first consider a nondimensional coordinate system (x,y,t) which has been nondimensionalized with respect to the sound speed a_0 of the polytropic gas at rest ahead of the shock and a suitable length scale L in the problem and introduce shock ray coordinate system (ξ,t) such that ξ = constant represents a shock ray (which for a shock moving into a polytropic gas at rest is orthogonal to the successive positions of the shock) and t = constant represents the shock front. Let M be the Mach number of the shock, i.e. the shock velocity divided by the constant speed of sound in the gas ahead of it), then Mdt represents an element of length along a ray. Let G be the metric corresponding to the variable ξ, i.e. $G\,d\xi$ is an element of length, say l along the shock front at a time $+t$. We can measure l from a suitable point on the wavefront, say the point where the ray $\xi = 0$ meets the wavefront. Then

$$l = \int_0^\xi G\,d\xi. \tag{1}$$

We choose the origin of the (x,y)-plane to be a suitable point on the initial shock front and ray $\xi = 0$ to be the one which starts from $(0,0)$ at $t = 0$. Let Θ be the angle that the normal to the shock front makes with the x-axis and N be the normal derivative of the fluid density just behind the shock. Then an approximation of the compatibility conditions of Ravindran and Prasad, for the propagation of a two-dimensional shock lead to a system of equations (see Prasad,[3] equations

(5.73–5.77)). We can show that a physically realistic conservation form of these are

$$(G \sin \Theta)_t + (M \cos \Theta)_\xi = 0 \tag{2}$$

$$(G \cos \Theta)_t - (M \sin \Theta)_\xi = 0 \tag{3}$$

$$(G(M\text{-}1)^2)_t + 2G(M\text{-}1)^2 N = 0 \tag{4}$$

$$(G(M\text{-}1)^4 N^{-2})_t = 0. \tag{5}$$

Equations (2)–(5) form the system of conservation laws which with appropriate initial values can be solved numerically in (ξ, t) plane. The solution can be mapped on to the (x,y)-plane giving the successive positions of the shock front and distribution of amplitude on them by numerically integrating the two equations.

$$X_\mathrm{t} = M \cos \Theta, \qquad Y_t = M \sin \Theta. \tag{6}$$

A total variation bounded (TVB) scheme based on the Lax–Friedrichs flux has been used to solve this system of conservation laws. The source term in the equation was handled through Strang splitting.[4] Effect of varying initial curvature as also the effects of varying the initial Mach strength M normal derivative N on the formation and propagation of kinks has been studied.

3. Results and discussion

Case (i): Propagation of a shock front initially parabolic in shape

The initial shock front is taken as

$$y^2 = bx, \qquad 1 \le b \le 8, \qquad |\,y| \le z \tag{7}$$

extended on either side as straight lines for $|y| >$ z by the tangents. M_0 is prescribed on the initial front as a symmetric function of Θ_0. We take

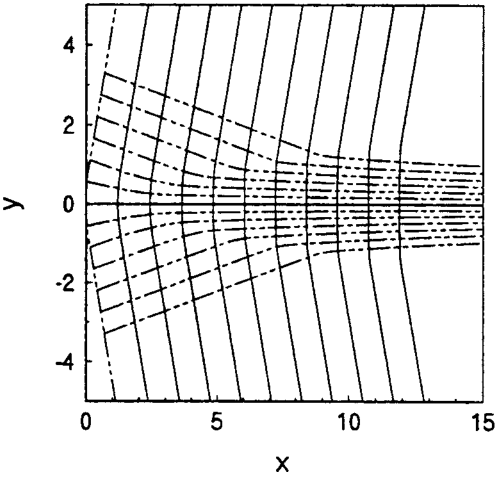


FIG. 1. Successive positions of shock front with rays and kinks shown by dots.

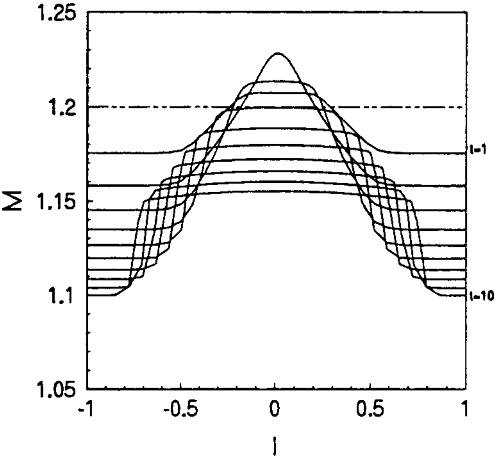


FIG. 2. Distribution of M with l.

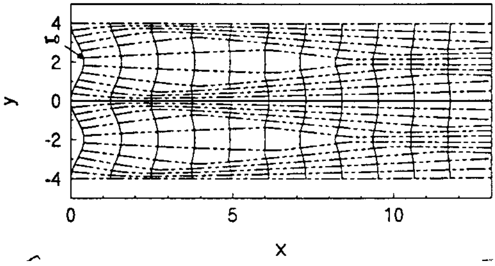


FIG. 3. Successive positions of an initially periodic shock front and associated rays from $t = 0$ to $t = 10$

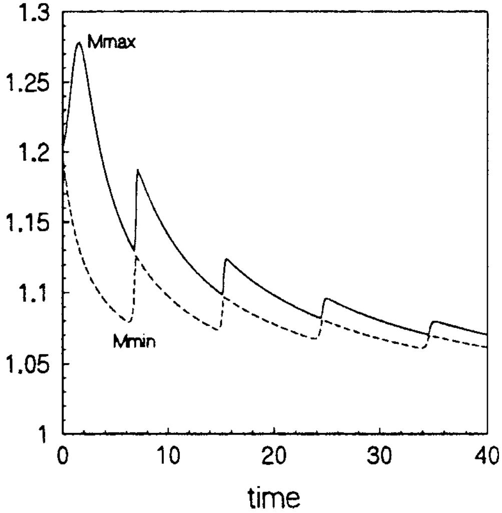


FIG. 4. Graphs of $M_{\max}$ (t) and $M_{\min}$ (t).

$$M_0 = \alpha \exp(-\beta\ \Theta_0^2) \tag{8}$$

where the parameter α is a measure of the strength of the initial shock front and β a measure of the rate of change of M_0 along the shock front. We also prescribe the initial value of N. For most cases, we have taken N_0 to be a constant and have varied it from 0.1 to 1.0.

The graph for one of the cases is shown in Fig. 1 for $b = 8$, $z = 1$, α 1.2, $\beta = 0$. Pairs of kinks appear on the front after a critical time, and the separation between them keeps on increasing as t increases. The Mach number increases in the curved part of the front till critical time when the kinks appear, after which it starts decreasing. But on the constant part, it keeps decreasing with t (Fig. 2).

Case (ii): Propagation of a shock front with initially sinusoidal shape and periodic amplitude distribution.

We now consider the initial shock front to be of a periodic sinusoidal shape

$$\mathrm{x}=0.2-0.\ 2\,A\cos\,(\pi y/B) \quad \text{where } A=1,\ \ B=2 \tag{9}$$

and M_0 is prescribed as in (8) with $\alpha = 1.2$ and $\beta = 0.0$. We have considered $N_0 = 0.1$ in this case. Figure 3 shows the shock fronts and the associated rays at successive times. With increasing time, the rays become straight line parallel to the x-axis and the shock front tends to become planar. In this case, the convergence and divergence of the rays are seen quite clearly.

A pair of kinks appear in each period of the shock front $t \approx 2$. Two kinks suitably paired first move closer to one another, interact producing a new pair of kinks which move apart and the process continues. After a long period of time, the amplitude M–1 decays to zero along each ray.

We define $M_{\max}$ (t) $(M_{\min}$ $(t))$ as the maximum (minimum) Mach number attained by the front at a given time t. $M_{\max}$ (t) attains a maximum value greater than $M_{\max}$ (0) at $t = 4$, which is approximately the time when the kinks first appear. Similarly, $M_{\min}(t)$ attains its minimum value at $t = 6$, the time when the kinks first interact. $M_{\max}(t) - 1$ and $M_{\min}(t) - 1$ both $\to 0$ and $M_{\max}(t) - M_{\min}(t) \to 0$ as $t \to \infty$ (Fig. 4).

References

1. Sturtevant, B. and Kulkarni, V. A. The focusing of weak shock waves *J. Fluid Mech.*, 1976, **73**, 651–671.
2. Prasad, P. and Sangeeta, K. Numerical simulation of converging nonlinear wavefronts, *J. Fluid Mech.*, 1999, **385**, 1–20.
3. Prasad, P. *Propagation of a curved shock and nonlinear ray theory*, Pitman Research Notes in Mathematics, Series 292, Longman Scientific and Technical, 1993.
4. LeVeque, R. J. and Yee, H. C. A study of numerical methods for hyperbolic conservation laws with stiff source term, *J. Computational Phys.*, 1990, **86**, 187–210.