SELECTION OF CONTINGENCY CASES IN PLANNING STUDIES

In most planning studies steady-state contingencies are assessed sequentially, simulating outages of one or more generating units and transmission facilities to investigate their effects on bus voltages and line power flows. A number of fast computer techniques have been developed for the analysis of such contingencies. They are well documented in a recent survey.\(^1\) Despite the speed of these methods, however, it is very costly and time-consuming to include all conceivable contingencies, especially when all primary contingencies are coupled with additional levels of secondary contingencies. On the other hand, selecting cases on the basis of the planner's experience and intuition can easily omit some critical ones.

The best approach is to survey all possible contingencies, but to do so by first ranking them in accordance with a relatively simple severity index. This can be done very efficiently by using Tellegen's theorem.\(^2\) The contingency ranking can be accomplished by ordering a set of normalized sensitivities from the greatest to the least. Nonlinear ac load flow equations are used to evaluate the effects of transmission and generation outages on a voltage quality system performance index, and a dc load flow model is used to rank contingencies with respect to real power flows. This method does not explicitly indicate whether a contingency is going to give bus voltage or circuit overload problems, but it indicates the severity of each contingency relative to the others. Full ac load flows must still be carried out—only on the "problem" cases. Furthermore, what constitutes the set of "problem" cases is easily determined by simply running the load flow for each case starting at the top of the list and stopping when the cases do not give problems. The cases below the stopping point need not be changed since they are already ranked in severity below the cases at the stopping point. In this manner, the method constitutes an adaptive contingency processor since the number of cases solved will vary depending upon system conditions. Such an adaptive contingency process is shown in Figure 1.

A contingency ranking algorithm based on Tellegen's theorem is used for single outage contingency cases in PTI's Bulk Power Adequacy Assessment Program PCAP. PTI has recently extended the method to common-mode, multiple-outage contingencies. In addition to its use in planning studies, the same contingency selection procedure is being incorporated into PTI's PSS system for energy control center applications where it is important to reassess a broad field of contingencies very quickly following a first contingency outage.

SHAFT TORQUES CAN BE A SERIOUS PROBLEM

Rotating masses and shafts representing the sections of the turbine and generator comprise elastic structures which, under impact loadings, can develop transient torques well in excess of rated steady state values. In the U.S., concern over the ability of turbine generators to withstand impact torques arose first in connection with series capacitor compensated transmission. Continuing investigations reveal that the problem deserves close attention in systems without series capacitors as well as further, that there is considerable misunderstanding over what the most severe cases are.

Impact torques capable of exciting shaft natural frequencies arise from changes in generator electromagnetic torque. The torque impact can have a direct component due to changes in 60 Hz generator current as well as oscillating components due to components of generator current at frequencies other than 60 Hz (dc and negative sequence components of fault current), and subsynchronous components in the case of series compensated systems.

Although it is possible to develop torque amplifications due to resonance between oscillating electrical torque frequencies and shaft torsional natural frequencies, such coincidence of frequencies is unusual. The principal mechanism for developing excessive shaft torques lies in sudden changes in normal power frequency current as can occur during faults, full load trips, reclosing or imperfect synchronization.

The torque augmentation in shaft sections can result from beat phenomena whereby very lightly damped modes of oscillation go through periodic coincidence of amplitude swings, and also from repeated impacts timed to coincide with peaks of shaft torque oscillations. The timing of the repeated impact is critical in that it can produce essentially doubling of the torque oscillations, as is illustrated in Figure 1.

The results of typical studies suggest that impacts produced by unit tripping situations are at least as severe as those produced by single line-to-ground faults. But reclosure into 3-phase faults can be even more severe. Because the damping of shaft oscillations is low, significant torque augmentation can occur when such switching operations are performed at intervals as long as several seconds apart.

In the absence of impact strength data for generator shafts,
MINICONPUTER ARCHITECTURE: Advice For Buyers

The achievements of modern minis and microcomputers are truly astounding, and more and more people are jumping onto the bandwagon. The costs of computation continue to decrease while capabilities of the computers continue to increase at a breathtaking pace. Consequently, more and more firms are now using minicomputers, not only for “mini” jobs but for some really tough jobs, including large engineering programs. Some of these experiences were painful—some pleasant. The main lesson learned was that most minicomputer systems are not suitable for large programs. Yet there are not many suppliers who will admit that his computer cannot do what his competitor’s can. It is very easy to be confused by advertising and brochure claims, particularly if one is inexperienced in the field. Some of the critical features needed for an efficient solution of large-scale problems are the most subtle. This article outlines some of the most important features that computer users need for large-scale programs and some of the more common pitfalls in application.

Computer Size and Speed

Size and speed are the most important considerations (although certainly not the only ones) in computer selection. The program size limitation is probably the most confusing issue for those not familiar with minicomputer applications. Program size limit is determined by the word size of the main memory and the computer’s internal registers. The most common size in today’s machines is 16 bits. If a 16-bit word is used as an address, it can access only $2^{16} = 65,536$ or (64K) unique locations. In fact, a 16-bit word is frequently the limit of program many machines. However, some machines access memory with a resolution of 8 bits, others reserve one bit of the address word for other purposes. In either of these cases, the effective addressing range is halved to 32K, 16-bit words.

There are several ways to get around this problem (short of increasing the word size), but the gains in some cases are illusory. One can, for example, simply use two 16-bit words to form the address. This allows an addressing range of $2^{32}$ (or about 4 billion) which would sem to be more than enough. However, the overhead of carrying these extra words in the program makes the program itself much bigger and causes it to execute very slowly. It is therefore not surprising that relatively few machines use this technique.

An increasingly popular method is called memory mapping, i.e., making use of a hardware algorithm to calculate a physical memory address from a program’s “virtual” address. Using this technique, there is no requirement for the “physical” address range to be the same as the “virtual” address range. Thus, the vendors are able to offer minis with extremely large memories which make use of mapping. But here again, “let the buyer beware!” Most such systems only allow the main CPU to be partitioned over several smaller “virtual” address ranges. In addition, the size restriction of any particular program is the same old 64K (or 32K) limit that it was before. Many prospective customers are misled on this point. Only one mini (of which we are aware), maps the address. This means that the address. This method is frequently mentioned as a “feature.” Most such systems which make use of mapping. But here again, “let the buyer beware!” Most such systems only allow the main CPU to be partitioned over several smaller “virtual” address ranges. In addition, the size restriction of any particular program is the same old 64K (or 32K) limit that it was before. Many prospective customers are misled on this point. Only one mini (of which we are aware), maps the address. This means that the address. This method is frequently mentioned as a “feature.”

Program overlaying is also frequently mentioned as a “cure” for addressing problems, especially in mapped machines which might allow several overlays to be stored in memory rather than on disk. However, most large programs are already overlayed and the effort required to squeeze them into even smaller overlay sections goes up exponentially.

Speed

Speed of execution of a program is usually important for engineering programs. Running a benchmark program is not very interesting for others. The program size limitation is probably the most confusing issue for those not familiar with minicomputer applications. Program size limit is determined by the word size of the main memory and the computer’s internal registers. The most common size in today’s machines is 16 bits. If a 16-bit word is used as an address, it can access only $2^{16} = 65,536$ or (64K) unique locations. In fact, a 16-bit word is frequently the limit of program many machines. However, some machines access memory with a resolution of 8 bits, others reserve one bit of the address word for other purposes. In either of these cases, the effective addressing range is halved to 32K, 16-bit words.

There are several ways to get around this problem (short of increasing the word size), but the gains in some cases are illusory. One can, for example, simply use two 16-bit words to form the address. This allows an addressing range of $2^{32}$ (or about 4 billion) which would sem to be more than enough. However, the overhead of carrying these extra words in the program makes the program itself much bigger and causes it to execute very slowly. It is therefore not surprising that relatively few machines use this technique.

An increasingly popular method is called memory mapping, i.e., making use of a hardware algorithm to calculate a physical memory address from a program’s “virtual” address. Using this technique, there is no requirement for the “physical” address range to be the same as the “virtual” address range. Thus, the vendors are able to offer minis with extremely large memories which make use of mapping. But here again, “let the buyer beware!” Most such systems only allow the main CPU to be partitioned over several smaller “virtual” address ranges. In addition, the size restriction of any particular program is the same old 64K (or 32K) limit that it was before. Many prospective customers are misled on this point. Only one mini (of which we are aware), maps the address. This means that the address. This method is frequently mentioned as a “feature.” Most such systems which make use of mapping. But here again, “let the buyer beware!” Most such systems only allow the main CPU to be partitioned over several smaller “virtual” address ranges. In addition, the size restriction of any particular program is the same old 64K (or 32K) limit that it was before. Many prospective customers are misled on this point. Only one mini (of which we are aware), maps the address. This means that the address. This method is frequently mentioned as a “feature.”

Program overlaying is also frequently mentioned as a “cure” for addressing problems, especially in mapped machines which might allow several overlays to be stored in memory rather than on disk. However, most large programs are already overlayed and the effort required to squeeze them into even smaller overlay sections goes up exponentially.

Speed of execution of a program is usually important for engineering programs. Running a benchmark program is not very difficult, but even here some pitfalls exist. First, there is no rational reason for a user to segregate the performance of the hardware from the performance of the combined hardware/software system. Indeed, the structure of some of the best machines makes it rather hard to tell the difference between hardware and operating system software. One should be sure that his benchmark is executed in the language (such as FORTRAN) and in the operating system which will ultimately be used. Several machines have excellent hardware performance but very poor software. One should also do some hard thinking about how representative the benchmark is. Many users tend to “load” their benchmarks with intensive floating point number crunching—the “heart” of their application. Vendors are interested in the size of these programs and many customers allow their machines to be strong at “benchmarking.” Actually, input and output operations have a much bigger impact on many programs than many users realize or care to admit. It is important to include these operations in a benchmark. Some of the fastest number crunchers are poor at I/O.

Other Considerations

Suppose a given computer is both big enough and fast enough for an engineering program. It is also wise to explore the system’s adaptability for program development work. Presumably, a user, after all, is interested in being involved in the effort to get existing applications up on the new machine. This effort must be considered as part of total cost. Unfortunately, the majority of minicomputer systems are poor in this regard. It is a good idea to refuse to accept a benchmark program in advance and to watch over his shoulder as his best man attempts to get the benchmark program adapted to his machine. Better yet, make him explain how to do it yourself! Many systems are extremely difficult to use and others are insecure. Consider some of the characteristics of the following systems which we have learned from actual use:

- **System A** is a multi-user system. However, bugs in one user’s FORTRAN program cause the entire system to crash, thus losing the work of all users. Any system which is crashable by a user level program is totally unacceptable.
- **System B** is a single-user system. It is oriented around batch job type work with job control commands such as JOB and COMPIL. In this particular system if the user mistakenly puts two COMPIL jobs in the same job the software will destroy the source file directories. Thus a simple mistake can cause much more than a lost job; it can destroy perhaps week’s worth of work spent to create the files.
- **System C** is oriented around batch and keyboard input. However, should the user type two characters in succession at a rate faster than the system is ready to accept, it crashes the system and erases the disk.
- **System D** is extremely slow at compiling and linking a FORTRAN program. It also requires contiguous disk space while at the same time partitioning unused disk space into many noncontiguous fragments. Its favorite trick is to first spend a long time in the compile and load phase then abort because there is not enough disk space. This kind of non-productivity can increase program development costs by several hundred percent.

Conclusions

If you are considering putting large engineering programs on a minicomputer, make the computer selection carefully. Do not depend on the “stature” of the manufacturer’s name alone. It is very unlikely that a very large share of any one vendor’s customers have the same requirements as you. The largest market for minicomputers is for real time applications which are characterized by needing to run large numbers of small programs at fast repetition rates. Not surprisingly, these real time systems are very poor for program development. Also the largest customers are the OEM’s who develop software once and duplicate the hardware many times. Development costs are not as important to them, and even then some of the most successful OEM’s write their own operating systems from scratch. The benefits of selecting the right computer are well worth the time it takes to be a critical judge.

R. J. Mills, Senior Engineer

PTI

SOFTWARE & HARDWARE

Separate or Common Contract?

Many suppliers of computer programs to the utility industry (PTI among them) have separate development programs completely separate from any of their supporting hardware requirements. Where hardware is required, quotations just cite the software requirements. Software and hardware are then the subject of separate contracts.

Some utilities now prefer purchasing both under one contract. Getting investment tax credit for the software is one incentive. There is some question concerning the availability of tax credit where software is acquired separately—especially if it is leased. But according to a 1971 IRS ruling, “software” which qualifies for tax credit includes software if the price charged for it is not stated separately from the cost of purchased computer hardware. This will become increasingly important as the ratio of hardware costs to software costs continues to decline.

Furthermore, it has been correctly observed that a combination hardware/software package is really a simulator—not a computer. This is an important distinction for some organizations which have purchasing policies. Accordingly, PTI will now quote, as an option, software and the supporting CPU’s and peripherals for the PSS/E program.

HIGH PHASE ORDER (HPO) —
An Alternative to UHV?

In the June 1975 PTI Newsletter (Issue No. 1), high phase order transmission was explained and compared briefly to conventional 3-phase lines. The idea appears to be gaining interest particularly as an alternative to UHV.

A recent PTI study, funded by the U.S. Department of Energy, spanned voltages from 138 kV through 800 kV. It appears that 138 kV is a logical voltage for any future experimental testing regardless of the voltage at which HPO lines might be applied.

All of the studies assumed a circular array as shown in Figure 1.

![Figure 1](image)

**FIGURE 1**

The principal conclusions of the study were as follows:

Impedance

As the number of phases increases, positive sequence reactance decreases. The positive sequence reactance of the 6-phase array of Figure 1 is approximately the same as for a double-circuit 3-phase line of the same voltage, recognizing that the latter would require increased phase spacing. Increasing to 12 phases would reduce the positive sequence impedance to 70 percent of the 6-phase value. Even though the 12-phase option would occupy no more space, the reduction in reactance might not be worth the added complexity of 12-phase conversion. It is true that a 12-phase line would have twice the thermal capability, but that same thermal increase could be achieved with larger conductors on the 6-phase option.

Fault Overvoltages

As phase order increases, the X0/X1 ratio of the line also increases. For the configuration of Figure 1, X0/X1 ratios are approximately 5, 7.8, 10.6, and 13 for phase orders of 3, 6, 12, and 24, respectively. For these ratios, the fault overvoltages are 1.35, 1.53, 1.45, and 1.35 p.u. respectively. While the fault overvoltages at 6 and 12 phases are somewhat higher than for 3-phase, the levels are not excessive and can be accommodated by typical equipment designs.

Unbalanced Currents

Load flow across 80 miles of 6-phase transmission line, both with and without array rotation, was compared with a 3-phase double-circuit configuration to evaluate unbalance. The line current for the untransposed 3-phase line was about 8 percent of the system assumed. A 6-phase line with no conductor rotation produced a current unbalance of less than 0.1 percent; and complete rotation of the 6-phase line reduced the unbalance to about 0.02 percent. The negative sequence currents in the generators were approximately 0.007 percent (one third the generator line) for both the 3-phase and 6-phase untransposed lines. Rotation of the 6-phase line reduced this to 0.0009 percent.

Radio Noise

As phase order is increased, holding conductor size constant, the conductor gradients and consequent radio noise levels decrease. In one comparison, the RI for a 3-phase 138 kV line was 8.6 dB higher than for the 6-phase option of corresponding phase-to-ground voltage.

Audible Noise

Audible noise is also significantly reduced for HPO. For the same line dimensions as the radio noise example, the audible noise in heavy rain for the 6-phase alternative is approximately 12 dB less than for the 3-phase case.

Electric Fields

Electric field strengths at ground level increase with an increase in phase order. For equivalent configurations, the maximum electric field intensity at ground level is 0.4 kV/m for a 3-phase 138 kV line, 1.0 kV/m for a double-circuit 138 kV line, and 0.1 kV/m for a 6-phase line. While the actual values for the 138 kV example are low, electric field strength at higher voltages could be a limiting design criterion.

Switching Surges

The phase-to-ground surge magnitudes, for equivalent system conditions, are not affected by increasing the phase order; but phase-to-phase surges increase. For example, line energization without surge limiting breakers produced phase-to-phase surges 50 percent greater in per unit of phase voltage for a 6-phase line than for a 3-phase line. However, addition of resistor preinsertion in the breaker reduced phase-to-phase surges in the 6-phase case from 3.0 p.u. to about 1.5 p.u., which allows even the most compact phase spacings.

Overvoltage prediction is easier than prediction of gap strength, however. There is virtually no available information on gap strength for long parallel conductors or on the effect of charges on the other phases on conductor-to-ground breakdown. Limited data for 3-phase configurations suggest this could be important in setting of phase spacing.

Lightning

Lightning performance of a 6-phase line was calculated to be comparable to typical double-circuit steel lattice 138 kV 3-phase lines. While the conventional tower, being higher and wider, will intercept more strokes, the high phase order line will have a larger number of phase-to-phase flashovers.

**Insulation Requirements**

Careful insulation coordination studies were conducted for 3-, 6-, 12-, and 24-phase systems and system phase-to-ground voltage levels of 80 kV through 442 kV. The insulation levels for the various high phase order stations are comparable to those required for 3-phase stations. In some cases, HPO stations required slightly more insulation.

Summary

It remains to be seen whether HPO transmission will see application, but its prospects are certainly brighter than they were two years ago. There appear to be two initial candidate applications for use on power systems:

**Uphasing**—Many 3-phase double-circuit tower lines in the U.S. are capable of uprating to a higher voltage from the standpoint of insulation and clearances but are limited by conductor size. If one converts to 6-phase, it is possible to raise the phase-to-ground voltage without increasing conductor gradient at all. This kind of uprating may be worthwhile to reduce losses even if the higher transfer capability of the HPO line is not used. The higher capability (or lower losses) is purchased at the price of reducing a double circuit to a single (6-phase) circuit. But the line does not have to be very long in order that the loss savings more than pay for the new terminal equipment.

**UHV Substitute?**—Since a 6-phase line is equivalent electrically to a double-circuit 3-phase line of the same phase-to-ground voltage, it is logical to suggest a 6-phase 800 kV line as an alternative to a 1200 kV line. Figure 2 compares the two configurations. Although the 800 kV HPO option would require less total MW than equivalent RI and audible noise performance, it is likely that losses would control the conductor size decision in the HPO case.

**REFERENCES**


Shaft Torques

(Continued from page 1)

It is important that utilities review torque levels expected in new operating situations relative to those known to have been successfully experienced by generators in the past. Problems may arise either because of a rise in the level of applied torque impacts, or because of reductions in the strength margins of new generating units.

Impact torque levels are generally increased by reductions of system impedances and by the use of series capacitors, and hence must be expected to rise as the power system becomes more strongly interconnected. Another factor aggravating the problem is the decrease in generator inertia that has come with improved cooling methods. This increases the proportion of impact torque transmitted through the shaft sections.

The shaft torque problem can be analyzed by comparing torques calculated for older units, during a typical disturbance, with those to be expected in new units in the same context. PTI's PSS/E program can now be used for that purpose, using network representations as in stability studies but with expanded machine models which include shaft dynamics.

A detailed paper on the shaft torque problem has been submitted for presentation at the 1978 Winter Meeting of the Power Engineering Society (IEEE).

J. M. Undrill, Senior Engineer
L. N. Hannett, Senior Engineer

REFERENCES
2. PTI Bulletin No. 91 — PSS/E Interactive Simulator for Engineering and Planning.

CONTIBUTORS

Louis N. Hannett received a Bachelor of Science degree in Electrical Engineering from Clarkson College of Technology. Since joining PTI shortly after graduation, he has worked in areas of dynamic simulation and control of power systems, having undertaken a number of studies of dynamic stability, load rejection, sub-synchronous resonance, and gas path dynamics for boiler-impeller interaction.

Richard J. Mills, with PTI for five years, has been very active in the fields of power system dynamics and real time control of power systems. His experience includes studies of load rejection overvoltages, multi-machine stability analysis, and the development of prime mover and energy supply system models for use in these studies. Other assignments have included software development for real time factory checkout of turbine controls, for digital control applications, and for the Dresden BWR Operator Training Simulator.

D. D. Wilson joined PTI in 1974 as Principal Engineer for Experimental Programs. He is presently responsible for overhead transmission and distribution experimental projects and the company's Saratoga Research and Development Center facility. He is active in several committees of IEEE and IEC. Previously with BNR, Mr. Wilson served as program manager for cooperative T&D research between GE and the power industry. He is a U.S. technical advisor to IEC TC 28 (Innovation Committee).

Bruce F. Wollenberg joined PTI in 1974, as a Senior Engineer engaged in development of energy control center software, system adequacy assessment techniques, and a variety of applications including optimization methods. He was formerly with the Westinghouse Research and Development Center, handling on-line load flow and security dispatch optimization applications. He has participated in feasibility studies for several large power pools and utility control centers, and was subsequently responsible for development and field checkout of security analysis programs.

NOTE

PTI's interactive planning program, PSS/E, described in the March 1976 issue of the Newsletter, is on its way to the best-seller list. The latest installation, in Norway, will soon be followed up by two more in the U.S. This will bring the number of PSS/E and PSS/2 installations to 23, including current letters of intent.

PTI has been awarded a contract by the Electric Power Research Institute (EPRI) to develop a procedure for efficient allocation of fuel for a group of interconnected utilities. The fuel dispatch time period will vary from a few hours to a week. Fuel allocations are to satisfy all operating requirements while optimizing a cost function. New York Power Pool is also a participant in this project.