PERFORMANCE MONITORING AND INTELLIGENT TRAINING

by

Matt Narotam, PhD
Supervisor, Software Systems
Burtek, Tulsa, OK 74101

and

Ms. Donna Behnke
Senior Systems Engineer

ABSTRACT

This paper presents a sophisticated approach to performance monitoring and Instructor/Operator Station (IOS) software utilizing an intelligent system for controlling the training scenario. This approach, based on existing software, provides the capability for a high level of student performance monitoring utilized successfully in complex maintenance trainers. This paper also discusses issues relating to peformance monitoring and organization of lesson plans to satisfy training requirements per student as identified by the system software. The system will control normal instructor functions including malfunctions, insertions and parameter modification, and will provide both visual and aural cues where necessary to aid in the clarification or augmentation of information presented to the trainee.

Introduction

The use of simulators as cost-effective training devices is commonplace in establishments providing training for personnel in the utilization of expensive equipment. The types of training accomplished with training devices ranges from simple familiarization with simulated systems and equipment, to developing individual student high-order skills.

There is a growing demand for increased performance assessment during simulator training and for tailoring of training scenarios to increase student proficiency in areas where the student appears deficient. In addition, there is a trend towards reducing demands placed on instructors in training scenarios which may require several instructors per simulator for multiple shift operation. In almost all training scenarios, there is a requirement for one or more instructors to operate and control the use of the device while training the students. The instructors are required to monitor student's performance and to tailor the exercise to eradicate deficiencies in student performance. In group training, where the instructor has the responsibility for training a class of students, the instructor is required to monitor each individual's performance and direct the training curriculum to complete training for individuals in an optimal manner.

The requirement to attend to each student's training program is a demanding burden which has traditionally been lessened by increasing the instructor/student ratio. Further complication occurs where several identical training devices are networked to a single IOS. In such cases, each trainer is capable of independent operation to the extent that different training exercises may be performed individually by each student. Under these conditions, the need for the instructor to provide the flexibility and concentration required for effective individual training is demanding. To meet this demand, there is an evident requirement for more effective use of automated procedures for controlling the individual training program.

Operation of automated procedures will not completely replace the function of instructors, but will reduce the demands placed on instructors and thereby decrease the instructor/student ratio.

Approach To Automated Training

This approach is based on a methodology developed and implemented by Burtek on several training devices. The methodology involves monitoring student performance during a training session and logging student errors. The major functions of an automated training system are:

- Monitor and record student performance,
- Assess student performance, and
- Intelligently select the next appropriate student exercise.

Performance Monitoring

The performance monitoring function requires detection and classification of erroneous student actions. The trainer response to the erroneous action may vary, depending on the training strategy, from "no action" to "trainer freeze", to "freeze the trainer until all incorrect actions are corrected in a self-teaching mode". The major requirement during performance monitoring is to establish a mechanism identifying all incorrect actions performed by the student. The approach to establishing this mechanism is to identify all correct actions which may be performed at each step of the training exercise. The aim should be to identify all actions relevant to the training scenario, to discard all nonessential steps, and to identify those incorrect actions which are hazardous or indicate the student is diverging from problem solution.

Performance Assessment

The performance assessment function of the system will assess the student's performance against established criteria and identify readiness and deficiencies in specific areas. The data analyzed for performance assessment is recorded during the performance monitoring phase. The results of the performance assessment will be used by the system in exercise selection and will be available to the instructor.

Exercise Selection

The exercise selection function addresses selection of the next appropriate exercise based on previous student performance. The student's performance record will be compared with the desired training profile. The results of this comparison will serve as a basis for identifying the next exercise.

Implementation Consideration

The approach to implementation of an automatic training system requires detailed consideration of the following criteria:

- Provision of a means of recording and maintaining student performance and therefore establishing a student record filing system,
- Identification of performance assessment criteria,
- Establishment of a mechanism for classifying students to identify entry level performance and training objectives for each student, and
- Identification of areas of specialization.

Student Record File

A training session will start by keyboard entry of student identification data usually in terms of student name, class or group, and a unique identification number. This process will establish a student record file associated with each lesson program. The record file will contain data describing the student's training parameters. For new students, the instructor will create a record file containing these parameters:

- Performance assessment criteria,
- Entry rating,
- Target rating, and
- Specialization.

The training parameters will be dependent on the type of training which can be provided by the training device. For example, for Aircrew Training (ATDs), Devices training may be provided for pilots, copilots, and flight engineers, which will necessitate the identification of assessment criteria for each category of student. All selectable parameters will be programmed into the training device to permit instructor selection for initialization of a student record file.

Performance Assessment Criteria

Performance assessment criteria refers to a set of parameters on which performance will be assessed. This data will form the basis of a scoring system used for assessment of student performance on each exercise. The performance criteria scoring will identify weaknesses in student performance and as a result, provide a datum for selection of the next appropriate exercise.

Initial information about each student must be provided by the instructor to enable initial exercise selection, and normal exercise progression.

Entry Rating. The entry rating will identify the students entry level qualifications (novice to expert) and will serve to identify the group of exercises which will be performed by the trainee. For the purpose of this paper, three categories are identified:

1. Instructional/Procedural Orientation - For this category of exercises, the student is assumed to be a novice and has very little or no knowledge of basic operational requirements of systems or equipment. The exercises will provide aural and visual cues indicating actions for the student to perform. All appropriate responses resulting from the students actions will be displayed. The student is expected to complete all required procedures.

2. Some Instructional Requirements - For this category, the student would have graduated from novice status and is expected to be knowledgeable about the operation of simulated systems and equipment. This mode of operation will provide limited additional cueing (over simulator response) to assist students in more difficult areas of exercise. The student is expected to demonstrate knowledge of system operations and performance and will be assessed according to the criteria established in the record file.

3. No Instructional Requirement - In this mode of operation, the student is expected to fully understand the operation of all simulated systems and equipment. No cues will be provided other than those generated as simulator responses to student actions.

Training Objective. The target rating parameter identifies the final rating to be achieved by the student. The automatic lesson planning algorithm will direct the student training accomplishments towards this goal.

Specialization. The specialization parameter will identify the specialist areas of training with which the student is to be biased. Examples are: radar system operation, electronic warfare, etc. The lesson planning algorithm will direct exercise selections to these specialist areas.

Exercise Selection

The mechanism for automatic exercise selection will employ the following:

- The data generated from student performance assessment software which identifies the areas of student weaknesses.
- Previous student performance as identified in the student record file,
- The area of specialization, as identified in the student record file, and

- A directory which identifies exercises available on the training device. The exercises will include both normal and emergency or malfunction exercises.

Existing Burtek Methodology

Software developed by Burtek provides the following capabilities:

- Coding of operational procedures to form a data base identifying expected actions and erroneous actions,

  NOTE: Data for establishing this data base is obtained from user operational manuals and documents identifying system operation and performance.

- Real-time software for processing students input and generating outputs to the training device,

- Instructor/Operation Station (IOS) software to allow the instructor to initialize training sessions and monitor students progress on the training device, and

- Error logging software to alert when the student performs an incorrect action and control the return to normality and reactivation of the exercise.

The existing methodology provides the following components for instructor control to generate student exercises:

- A mechanism is provided whereby the instructor may generate lessons, each lesson consisting of any number of exercises. The instructor may edit, create or delete lessons.

- As part of the lesson set-up feature, the instructor may identify areas where prompting is required. The system permits the instructor to insert a message which would be displayed at the

- Also, as part of the lesson set-up feature, the instructor may specify new malfunction values for each exercise.

- The instructor may request a printout of student performance which may be kept in a student record file.

Exercise Building

A data base of exercises must be established, from which a lesson plan will be constructed for conducting a training session. The exercise data base will consist of procedures identifying the student's expected actions at any point in time during the training session, based on student's previous actions. The procedures will also identify the status of the trainer used for classifying any erroneous student actions. The procedures data base is generated using an English-like language called Modular Procedural Assembler (MPA). The heart of the MPA system is the symbol description data base. The data base is mapped onto COMMON I/O area that interfaces with the trainer hardware. The MPA System processes a coded procedure and produces a data file containing references to the I/O interface. The data file is evaluated by a procedure monitor to determine the expected input on the trainer and the responses to be displayed on the equipment. The MPA software is coded in FORTRAN 77.

Symbolic Data Base

The symbolic data base describes all components monitored and controlled by the MPA system, and enables the system to translate from component names sensible to the user, to a machine-oriented label and vice versa. This reverse process is important to allow the machine to generate information about a component in a form easily understood by the user.

The component description also enables various system states of the component to be translated into a format understood by either user or machine. The component description includes information enabling MPA to monitor the use of the component in the procedure and includes information required by the real-time part of the MPA system.

The structure of the symbolic data base allows the following description to be supplied for any component on the trainer:

Component I/O type:	Discrete I/O Analog I/O Real I/O Integer I/O Logical I/O Character I/O
Component Index:	Address in unpacked I/O table
Symbolic Label:	A label which is given to the component (64 characters)
Component Device Tape:	Push button Toggle Multiway Switch, etc.
Range/Tolerance:	-1.0 to +1.0 ± 0.01
Value/States:	Off, On

MPA Constructs

The following paragraphs present a brief description of major MPA constructs.

Await. The AWAIT statement causes the MPA software to await student action, such as pressing a switch or turning a device on. The component's expected state is stacked until the actual change is made by the student. The action is verified and then the next function in the procedure is performed. An example of a simple AWAIT construct is:

```
Await MENU on LDDI to ON;
```

The component awaited is the MENU push button on the left digital display indicator, abbreviated to LDDI.

The AWAIT construct can accommodate more complex structures such as:

```
Await MENU on LDDI to ON or
  MENU on RDDI to ON;
```

This construct allows the student to select either the left or the right digital display indicator in the procedure.

In circumstances where events may be awaited in any order, the following form may be used:

```
Await GROUND_POWER to ON and
  HYDRAULIC_POWER to ON;
```

Set. The SET statement allows a trainer output to be displayed. For example:

```
Set DISPLAY on LDDI to MENU;
```

The SET statement may be combined with the AWAIT statement, as in:

```
Await MENU on LDDI to ON:
Set DISPLAY to MENU;
```

Note that the system name, LDDI, associated with DISPLAY is not specified in the SET statement because MPA assumes the system name used in the AWAIT. Both AWAIT and SET may have been combined in the following manner:

```
Await MENU on LDDI to ON
  then_set DISPLAY to MENU;
```

The use of this feature is demonstrated with an example using the AND construct:

```
Await GROUND POWER to ON
  then_set GROUND_POWER_LIGHT to ON
and HYDRAULIC_POWER to ON
  then_set HYDRAULIC_POWER_LIGHT to ON;
```

In the above example, the lights will be displayed as the individual switches are activated. An output may be set to some time function as in: Set METER to 1.0 using RAMP over 2.0; which causes a meter to be ramped from 0.0 to 1.0 in two seconds.

Perform. The PERFORM statement is similar to a FORTRAN CALL statement. PERFORM is used to activate another procedure which resides in a library, for example:

```
Perform POWER_ON;
```

If. The IF statement allows the direction of code execution to be altered depending on the state of inputs or outputs on the trainer, for example:

```
If GROUND_POWER_SWITCH is OFF
  Perform POWER_ON;
Else
  Await GROUND_POWER_SWITCH to OFF;
  Perform POWER_ON;
End_if:
```

This example demonstrates the use of the IF statement to perform the correct action depending on the state of the GROUND_POWER_SWITCH.

The following example displays another use of the IF statement:

```
Await MENU on LDDI to ON
  then_set DISPLAY to MENU
or MENU on RDDI to ON
  then_set DISPLAY to MENU;

If DISPLAY on LDDI is MENU
  Perform LDDI_TEST;
else_if DISPLAY on RDDI is MENU
  Perform RDDI_TEST;
end_if;
```

These statements may be used to code steps in procedures where the student is instructed to select the LDDI or the RDDI to perform the test required.

The IF statement may also be used to perform alternate actions if a malfunction is active. The following example displays the use of the IF statement for that purpose.

```
Await SDRS to ON;
Set DISPLAY to "IN TEST";
If MF0052 is ON
  Set DISPLAY to "NO GO";
  Perform P00400;
  Restart;
Else
  Set DISPLAY to "GO" after 5 seconds;
End_if;
```

This example displays a procedure involving one of the display units. A pushbutton labeled SDRS is pressed and the normal response display is "IN TEST" on the screen, changing to "GO" after five seconds. Under the malfunction condition, the display changes to "NO GO", whereupon the student locates the cause of the malfunction by performing another procedure. The RESTART statement causes control to return to the beginning of the procedure after the malfunction is corrected.

Until. The UNTIL construct permits a block of code to be executed until some condition is satisfied.

Display. The DISPLAY statement permits a message to be displayed at an appropriate point in the exercise, which may be used to clarify a step or print a warning, etc.

Expansion of Burtek's Methodology

Burtek's current software approach provides the performance monitoring aspect of an automatic training system. This approach can be further developed to include assessment of the student's performance and automatic control of exercise selection.

Procedure Classification

The instructions in the procedure data base will be required to be classified into tasks serving to collectively identify procedures associated with specialist requirements. The grouping will distinguish procedures for normal system operation from that necessary to be performed under emergency or malfunction conditions. This information will be required during the phase of operation determining the training exercise to be performed by the student.

Task Matrix

Exercise selection for each student will be based on a previously defined task matrix. This matrix consists of a list of available precoded exercises along with which exercise should be used as remediation for each class of error possible within the exercise.

Error Classification

Classification of student errors is application dependent, but one method that may be used is to break the exercise into steps where any incorrect student action within each step will signal a particular class of failure. Software is provided to track the steps in which errors occurred. The MPA system contains a construct for identifying exercise steps.

Sample Exercises

This paper will develop a task matrix for a series of exercises in training a pilot in takeoff. It will be assumed that the step-wise method of error classification has been used in the exercise building. Table 1 shows takeoff exercises to be included in the task matrix, along with associated possible error classes. In addition to completing the exercise correctly, each exercise should be completed within an acceptable length of time. Failure to do so will be considered to indicate an unfamiliarity with the cockpit layout. This error class will be designated as TIME. Other error classes will be PROCEDURAL - failure to follow required procedure, WIND - failure to account for the effects of wind, and EMERGENCY - failure to respond appropriately to an emergency.

The exercise levels, given in the table, describe the amount of trainer cueing included in the exercise. An orientation level exercise will provide aural or visual cues for all actions to be taken by the student. Simulated aircraft responses will be displayed. Coached exercises will provide cueing in the more difficult areas of the exercise, in addition to simulated responses to actions. Simulation exercises will provide no cueing over simulated responses.

Normal Progressions. The task matrix for the takeoff exercises will consist of two parts. The first part, listed in Table 2, describes the normal progression through the exercises if no errors occur. For the novice student, all exercises are included, providing orientation, coaching, and simulation. The intermediate student will be presumed to be knowledgeable about the basic equipment and procedures, needing only coaching and simulation. The expert student only needs to complete the simulation exercises.

Remediation. The second part of the task matrix, presented in Table 3, describes the remediation to be provided, dependent upon the class of error in the attempted exercise and the previously failed exercises.

The general strategy in remediation is to regress to the next exercise level which will provide coaching in the error class area. Remediation determination is independent of the student expertise level.

The Student Record File

A record file will be maintained for each student containing student identification information, the student's entry rating, target rating, and specialization, and will record the results of all exercise attempts. The data in the file will be used in the selection of exercises, and will be available to the instructor for evaluation of the student's completed task.

Exercise Selection Data. Exercise selection is based upon the student's entry rating and successful completion of exercises and failed exercises. Since completion of all necessary exercises may well require more than one session on the training device, it is necessary to record exercise history between sessions. Necessary information would include:

- Last attempted exercise,
- Last attempted exercise error class,
- List of required exercises, and
- List of passed exercises.

Evaluation Data. For an instructor to evaluate how well a student completed his assigned task, the instructor should know the following information about each exercise available in the task:

- Exercise description,
- Whether the exercise was required,
- Number of attempts, and
- Number of each class of error.

Record Format. A possible student record format to satisfy the above requirements would be:

Section 1 - Permanent Data;

- Student Name,
- Student ID,
- Entry Rating,
- Target Rating, and
- Specialization - required task.

Section 2 - Task Data;

- Task Name,
- Number of exercises in task,
- Task completed flag,
- Last attempted exercise, and
- Last attempted exercise error class.

Section 3 - Exercise Results;

- A matrix of the following data for each exercise in the task,
- Exercise description,
- Exercise required flag,
- Number of successful attempts, and
- For each class of error in the exercise, the number of occurrences of that error.

This record should then function to provide both exercise selection data and student evaluation data.

TABLE 1 SAMPLE EXERCISES

EXERCISE NUMBER	EXERCISE DESCRIPTION	POSSIBLE ERROR CLASSES	EXERCISE LEVEL
1	TAKEOFF SWITCHOLOGY	NONE	ORIENTATION
2	SIMPLE TAKEOFF - PROCEDURE	TIME	ORIENTATION
3	SIMPLE TAKEOFF - COACHED	TIME, PROCEDURE	COACH
4	SIMPLE TAKEOFF - SIMULATED	TIME, PROCEDURE	SIMULATION
5	TAKEOFF WITH WIND - PROCEDURE	PROCEDURE	ORIENTATION
6	TAKEOFF WITH WIND - COACHED	TIME, PROCEDURE	COACH
7	TAKEOFF WITH WIND - SIMULATED	TIME, PROCEDURE, WIND	SIMULATION
8	TAKEOFF WITH EMERGENCIES - PROCEDURE	PROCEDURE	ORIENTATION
9	TAKEOFF WITH EMERGENCY 1 - COACHED	TIME, PROCEDURE	COACH
10	TAKEOFF WITH EMERGENCY 2 - COACHED	TIME, PROCEDURE	COACH
11	TAKEOFF WITH EMERGENCY 1 - SIMULATED	TIME, PROCEDURE,	SIMULATION
12	TAKEOFF WITH EMERGENCY 2 - SIMULATED	TIME, PROCEDURE, EMERGENCY	SIMULATION
13	TAKEOFF WITH WIND AND EMERGENCY 1 - SIMULATED	TIME, PROCEDURE, EMERGENCY	SIMULATION
14	TAKEOFF WITH WIND AND EMERGENCY 2 - SIMULATED	TIME, PROCEDURE, EMERGENCY	SIMULATION

TABLE 2 NORMAL EXERCISE PROGRESSION

EXERCISE NUMBER	EXERCISE DESCRIPTION	STUDENT LEVEL NOVICE	INTERMEDIATE	EXPERT	NEXT EXERCISE DESCRIPTION	NEXT EXERCISE NUMBER
1	TAKEOFF SYMBOLOGY	X			SIMPLE TAKEOFF - PROCEDURE	2
2	SIMPLE TAKEOFF - PROCEDURE	X			SIMPLE TAKEOFF - COACHED	3
3	SIMPLE TAKEOFF - COACHED	X	X		SIMPLE TAKEOFF - SIMULATED	4
4	SIMPLE TAKEOFF - SIMULATED	X			TAKEOFF WITH WIND - PROCEDURE	5
			X		TAKEOFF WITH WIND - COACHED	6
				X	TAKEOFF WITH WIND - SIMULATED	7
5	TAKEOFF WITH WIND - PROCEDURE	X			TAKEOFF WITH WIND - COACHED	6
6	TAKEOFF WITH WIND - COACHED	X	X		TAKEOFF WITH EMERGENCIES - SIMULATED	7
7	TAKEOFF WITH WIND -- SIMULATED	X			TAKEOFF WITH EMERGENCIES - PROCEDURE	8
			X		TAKEOFF WITH EMERGENCY - COACHED	9
				X	TAKEOFF WITH EMERGENCY - SIMULATED	11
8	TAKEOFF WITH EMERGENCIES - PROCEDURE	X			TAKEOFF WITH EMERGENCY - COACHED	9
9	TAKEOFF WITH EMERGENCY 1 - COACHED	X	X		TAKEOFF WITH EMERGENCY 2 - COACHED	10
10	TAKEOFF WITH EMERGENCY 2 - COACHED	X	X		TAKEOFF WITH EMERGENCY 1 - SIMULATED	11
11	TAKEOFF WITH EMERGENCY 1 - SIMULATED	X	X	X	TAKEOFF WITH EMERGENCY 2 - SIMULATED	12
12	TAKEOFF WITH EMERGENCY 2 - SIMULATED	X	X	X	TAKEOFF WITH WIND & EMERGENCY 1 - SIMULATED	13
13	TAKEOFF WITH WIND & EMERGENCY 1 - SIMULATED	X	X	X	TAKEOFF WITH WIND & EMERGENCY 2 - SIMULATED	14
14	TAKEOFF WITH WIND & EMERGENCY 2 - SIMULATED	X	X	X	TASK COMPLETE	--

TABLE 3 REMEDIATION DETERMINATION

ATTEMPTED EXERCISE #	ATTEMPTED EXERCISE DESCRIPTION	ERROR CLASS	ALSO FAILED EXERCISE #	REMEDIAL DESCRIPTION	REMEDIAL EXERCISE NUMBER
1	TAKEOFF SWITCHOLOGY	NONE	ANY	-	-
2	SIMPLE TAKEOFF - PROCEDURE	TIME	ANY	TAKEOFF SWITCHOLOGY	1
3	SIMPLE TAKEOFF - COACHED	TIME	ANY	TAKEOFF SWITCHOLOGY	1
		PROCEDURE	ANY	SIMPLE TAKEOFF - PROCEDURE	2
4	SIMPLE TAKEOFF - SIMULATED	TIME	ANY	TAKEOFF SWITCHOLOGY	1
		PROCEDURE	ANY	SIMPLE TAKEOFF - COACHED	3
5	TAKEOFF WITH WIND - PROCEDURE	PROCEDURE	ANY	SIMPLE TAKEOFF - PROCEDURE	2
6	TAKEOFF WITH WIND - COACHED	TIME	ANY	TAKEOFF SWITCHOLOGY	1
		PROCEDURE	7 or 13 or 14	TAKEOFF WITH WIND - PROCEDURE	5
		PROCEDURE	NOT 7 or 13 or 14	SIMPLE TAKEOFF - PROCEDURE	2
7	TAKEOFF WITH WIND - SIMULATED	TIME	ANY	TAKEOFF SWITCHOLOGY	1
		PROCEDURE	ANY	SIMPLE TAKEOFF - PROCEDURE	1
		WIND	ANY	TAKEOFF WITH WIND - COACHED	6
8	TAKEOFF WITH EMERGENCIES PROCEDURES	PROCEDURE	ANY	SIMPLE TAKEOFF - PROCEDURE	2
9	TAKEOFF WITH EMERGENCY 1 - COACHED	TIME	ANY	TAKEOFF SWITCHOLOGY	1
		PROCEDURE	11 or 13	TAKEOFF WITH EMERGENCIES - PROCEDURE	8
		PROCEDURE	NOT 11 or 13	SIMPLE TAKEOFF - PROCEDURE	2
10	TAKEOFF WITH EMERGENCY 2 - COACHED	TIME	ANY	TAKEOFF SWITCHOLOGY	1
		PROCEDURE	12, or 14	TAKEOFF WITH EMERGENCIES-PROCEDURE	8
		PROCEDURE	NOT 12 or 14	SIMPLE TAKEOFF - PROCEDURE	2
11	TAKEOFF WITH EMERGENCY 1- SIMULATED	TIME	ANY	TAKEOFF SWITCHOLOGY	1
		PROCEDURE	ANY	SIMPLE TAKEOFF - PROCEDURE	2
		EMERGENCY	ANY	TAKEOFF WITH EMERGENCY 1 - COACHED	9
12	TAKEOFF WITH EMERGENCY 2 - SIMULATED	TIME	ANY	TAKEOFF SWITCHOLOGY	1
		PROCEDURE	ANY	SIMPLE TAKEOFF - PROCEDURE	2
		EMERGENCY	ANY	TAKEOFF WITH EMERGENCY 2 - COACHED	10
13	TAKEOFF WITH WIND AND EMERGENCY 1 - SIMULATED	TIME	ANY	TAKEOFF SWITCHOLOGY	1
		PROCEDURE	ANY	SIMPLE TAKEOFF - PROCEDURE	2
		EMERGENCY	ANY	TAKEOFF WITH EMERGENCY 1 - COACHED	9
		WIND	ANY	TAKEOFF WITH WIND - COACHED	6
14	TAKEOFF WITH WIND AND EMERGENCY 2 - SIMULATED	TIME	ANY	TAKEOFF SWITCHOLOGY	1
		PROCEDURE	ANY	SIMPLE TAKEOFF - PROCEDURE	2
		EMERGENCY	ANY	TAKEOFF WITH EMERGENCY 2- COACHED	10
		WIND	ANY	TAKEOFF WITH WIND - COACHED	6

Conclusion

This paper has presented an approach to providing an automated system for controlling student exercises on simulator training devices. The paper identified a mechanism for defining expected student actions which are compared with actual actions to determine the correctness of the action. Student's performance is monitored and assessed as part of the strategy to select the next appropriate exercise for each student.

This approach will require manually establishing a data base consistive of a pool of exercises from which the next exercise is selected. Additionally, each task to be learned will require an exercise matrix ensuring that the student has demonstrated proficiency to the required level in the assigned task.

Using this approach, the instructor need not be interrupted to select a new exercise whenever a student completes his assigned exercise.

About the Author

Dr. Matt Narotam is Supervisor of the Software Systems Group at Burtek with responsibilities for simulator software engineering including instructor station software, input/output software, executive software, operating system interface and new product applications. Dr. Narotam has developed a high-order language for coding training procedures at Burtek. Dr. Narotam has also developed graphics software and system models for F/A-18 trainers manufactured by Burtek. Dr. Narotam has over ten years experience in the field of simulation. Prior to joining Burtek, Dr. Narotam pursued the development of simulation techniques at the Computer Simulation Center, University of Sanford, England, UK. Dr. Narotam obtained his Ph.D. at the University of Sanford with a thesis on a Continuous System Simulation Language (CSSL).

Donna Behnke is a Senior Systems Engineer in the Software Systems Group at Burtek. Ms. Behnke's experience at Burtek includes development of software for instructor/operator stations and system models, specializing in human interface aspects of simulator software. Ms. Behnke has eleven years experience in training-related environment including five years spent on developing interactive software systems. Ms. Behnke received her Bachelor of Science Mathematics and a Secondary Education Certificate at the State University of New York, College at Fredonia and studied Computer Science and Engineering at California State University, Northridge, and Rogers State College, Claremore.