

Simulation Technology & Applied Research, Inc.

11520 N. Port Washington Rd., Suite 201, Mequon, WI 53092 Tel: (262) 240-0291 FAX: (262) 240-0294

Analysis Solutions for Electromagnetic Design

August 15, 2006

To:
Program Officer: Ingham Mack
SBIR Program Manager: Cathy Nodgaard
Contract Administrator: John Lewinski

Dear Sirs and Ms:

Enclosed please find Interim Report 2 for our Phase 2 SBIR project entitled "Computer-Aided Design and Optimization of High Performance Vacuum Electronic Devices", contract number N00014-05-C-0375. This report covers the period Feb. 1, 2006 – Apr. 30, 2006. Also please find SF-298 as required.

Best Regards,

JOHN F. DEFORD

John F. DeFord
President

P: (262) 240-0291 x102

F: (262) 240-0294

E: john.deford@staarinc.com



Simulation Technology & Applied Research, Inc.

11520 N. Port Washington Rd., Suite 201 Mequon, WI 53092-3432

P: 1.262.240.0291 F: 1.262.240.0294 W: www.staarinc.com

Computer-Aided Design and Optimization of High-Performance Vacuum Electronic Devices

SBIR Phase 2 Progress Report 2

(covering period 2/1/2006-4/30/2006)

Contract/Purchase Order Number: [N00014-05-C-0375](#)

Simulation Technology & Applied Research Report Number: [06-SBIR-ONR-T2](#)

Prepared for the Office of Naval Research

August 15, 2006

John F. DeFord and Ben Held
Simulation Technology & Applied Research, Inc.
11520 N. Port Washington Rd., Suite 201
Mequon, WI 53092

P: 1-262-240-0291 x102 F: 1-262-240-0294 E: john.deford@staarinc.com

John Petillo
Scientific Applications International Corporation
Suite 130, 20 Burlington Mall Rd.
Burlington, MA 01813

P: 1-781-221-7615 F: 1-781-270-0063 E: jpetillo@bos.saic.com

SBIR Data Rights

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Washington Headquarters Service, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.

1. REPORT DATE (DD-MM-YYYY) 21-02-2006		2. REPORT TYPE Interim		3. DATES COVERED (From - To) Feb., 2005 - Apr., 2006	
4. TITLE AND SUBTITLE Computer-Aided Design and Optimization of High-Performance Vacuum Electronic Devices				5a. CONTRACT NUMBER N00014-05-C-0375	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S) J. F. DeFord, B. Held, L. Chernykova Simulation Technology & Applied Research, Inc. J. Petillo Scientific Applications International Corporation				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Simulation Technology & Applied Research, Inc. 11520 N. Port Washington Rd., Suite 201, Mequon, WI 53092				8. PERFORMING ORGANIZATION REPORT NUMBER 06-SBIR-ONR-T2	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Office of Naval Research 875 North Randolph St. Suite 1425 Arlington, VA 22203-1995				10. SPONSOR/MONITOR'S ACRONYM(S) ONR	
				11. SPONSORING/MONITORING AGENCY REPORT NUMBER 001AC	
12. DISTRIBUTION AVAILABILITY STATEMENT Unlimited/Unclassified					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT Initial development of the multi-directional search (MDS) and differential evolution (DE) optimization methods is now complete, as is the initial user interface to these capabilities within the Analyst V8.1 software package that is presently in "beta" form. The 8-processor Linux cluster purchased for the project has been used extensively during the past few months to test the optimization system, and has shown excellent performance with floating point computation rates exceeding 4 GFLOPS on some problems. The cluster has also recently been upgraded to 64GB of system memory, allowing solution of significantly larger problems.					
15. SUBJECT TERMS optimization,LINUX cluster,differential evolution,multi-directional search					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT UU	18. NUMBER OF PAGES 6	19a. NAME OF RESPONSIBLE PERSON John F. DeFord
a. REPORT UU	b. ABSTRACT UU	c. THIS PAGE UU			19b. TELEPHONE NUMBER (Include area code) (262) 240-0291 x102

INSTRUCTIONS FOR COMPLETING SF 298

1. REPORT DATE. Full publication date, including day, month, if available. Must cite at least the year and be Year 2000 compliant, e.g., 30-06-1998; xx-08-1998; xx-xx-1998.

2. REPORT TYPE. State the type of report, such as final, technical, interim, memorandum, master's thesis, progress, quarterly, research, special, group study, etc.

3. DATES COVERED. Indicate the time during which the work was performed and the report was written, e.g., Jun 1997 - Jun 1998; 1-10 Jun 1996; May - Nov 1998; Nov 1998.

4. TITLE. Enter title and subtitle with volume number and part number, if applicable. On classified documents, enter the title classification in parentheses.

5a. CONTRACT NUMBER. Enter all contract numbers as they appear in the report, e.g. F33615-86-C-5169.

5b. GRANT NUMBER. Enter all grant numbers as they appear in the report, e.g. 1F665702D1257.

5c. PROGRAM ELEMENT NUMBER. Enter all program element numbers as they appear in the report, e.g. AFOSR-82-1234.

5d. PROJECT NUMBER. Enter all project numbers as they appear in the report, e.g. 1F665702D1257; ILIR.

5e. TASK NUMBER. Enter all task numbers as they appear in the report, e.g. 05; RF0330201; T4112.

5f. WORK UNIT NUMBER. Enter all work unit numbers as they appear in the report, e.g. 001; AFAPL30480105.

6. AUTHOR(S). Enter name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. The form of entry is the last name, first name, middle initial, and additional qualifiers separated by commas, e.g. Smith, Richard, Jr.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES). Self-explanatory.

8. PERFORMING ORGANIZATION REPORT NUMBER. Enter all unique alphanumeric report numbers assigned by the performing organization, e.g. BRL-1234; AFWL-TR-85-4017-Vol-21-PT-2.

9. SPONSORING/MONITORS AGENCY NAME(S) AND ADDRESS(ES). Enter the name and address of the organization(s) financially responsible for and monitoring the work.

10. SPONSOR/MONITOR'S ACRONYM(S). Enter, if available, e.g. BRL, ARDEC, NADC.

11. SPONSOR/MONITOR'S REPORT NUMBER(S). Enter report number as assigned by the sponsoring/ monitoring agency, if available, e.g. BRL-TR-829; -215.

12. DISTRIBUTION/AVAILABILITY STATEMENT. Use agency-mandated availability statements to indicate the public availability or distribution limitations of the report. If additional limitations/restrictions or special markings are indicated, follow agency authorization procedures, e.g. RD/FRD, PROPIN, ITAR, etc. Include copyright information.

13. SUPPLEMENTARY NOTES. Enter information not included elsewhere such as: prepared in cooperation with; translation of; report supersedes; old edition number, etc.

14. ABSTRACT. A brief (approximately 200 words) factual summary of the most significant information.

15. SUBJECT TERMS. Key words or phrases identifying major concepts in the report.

16. SECURITY CLASSIFICATION. Enter security classification in accordance with security classification regulations, e.g. U, C, S, etc. If this form contains classified information, stamp classification level on the top and bottom of this page.

17. LIMITATION OF ABSTRACT. This block must be completed to assign a distribution limitation to the abstract. Enter UU (Unclassified Unlimited) or SAR (Same as Report). An entry in this block is necessary if the abstract is to be limited.

Abstract

Initial development of the multi-directional search (MDS) and differential evolution (DE) optimization methods is now complete, as is the initial user interface to these capabilities within the Analyst V8.1 software package that is presently in “beta” form. The 8-processor Linux cluster purchased for the project has been used extensively during the past few months to test the optimization system, and has shown excellent performance with floating point computation rates exceeding 4 GFLOPS on some problems. The cluster has also recently been upgraded to 64GB of system memory, allowing solution of significantly larger problems.

Executive Summary

The following items have been accomplished during this period:

- Qualification of Linux cluster at STAR is finished. Cluster has shown performance above 4 GFLOPS on distributed LU factorization.
- Preliminary work on the Analyst user interface to the optimization library has finished, with the initial implementation fielded in beta form in version 8.1 of Analyst.
- Optimization capabilities have been successfully applied to multi-variate collector problems using MICHELLE, and also to driven-frequency optimizations using the Analyst DFEM solver.

Qualification of Linux Cluster

We have run many tests on the system as originally configured (with 4GB of system memory per processor), and it has proved to be quite stable. Although the 1GB interconnect is relatively slow (by typical cluster standards), for most problems it is not a significant factor in the overall performance of the system.

We recently upgraded the memory to 8GB per processor (bringing the total system memory to 64GB), and subsequent to the upgrade the system has been less reliable, with nodes going down on occasion. This appears to be due to a combination of factors that include one or more bad memory chips, and an air-conditioning problem that resulted in slightly warmer temperatures in the machine room during part of the testing period. We temporarily took 4 processors offline (still with a total memory of 32GB), while the remaining processors were either powered down or being used to test memory.

We are now in the process of sequentially testing all of the memory chips, and have also addressed the air-conditioning problem. The cluster is presently back to 6 processors/48GB memory, and we expect to have all eight processors at full memory within the next few days.

Optimization User Interface

Implementation of differential evolution, multi-directional search, and a gradient-based zero-crossing algorithm is now complete. The optimization controller can also be used to perform parameter sweeps in which the goal function is evaluated and reported for a set of parameter vectors in a user-specified range.

The basic user-interface to the optimizer is also complete, with the panels shown in our previous progress report now fully implemented. It consists of a set of “wizard” panels that walk a user through the setup process. The optimization process relies heavily on the embedded Python system: Python scripts are used to define the goal function and constraint functions (see Fig.1-2), and later we plan to add support for specifying a user-defined optimization algorithm in a Python script. User can edit the existing scripts, or add new ones as necessary.

Once the optimization is started, a dialog is displayed that shows the real-time progress of the process on several different tabs (see Fig. 3). The best case is retained, while all other cases are deleted after each iteration to avoid accumulation of unnecessary files and data.

Moreover, the complete history of the parameter vectors and resulting metric values for an optimization can be copied from the Experiment Details panel of the Status dialog and pasted into an Excel spreadsheet for review or further manipulation (such as curve-fitting of other analysis).

Early Experience with Optimization System

The optimization system has now been used on several different types of problems, and both MDS and differential evolution appear to be functioning properly. On multivariate problems for which there are a lot of local minima, differential evolution generally does a better job of finding an optimum value, but it takes longer to converge than MDS. Both approaches require a significant number of analyses, with problems involving several parameters often taking several hundred or more function evaluations. Since a single metric function evaluation can take a few minutes to a few hours time to perform, the overall time required to do an optimization can be days to weeks. Because of this, we are looking at ways to improve the efficiency of the algorithms, including the use of local functional approximations to the metric, and space mapping wherein low-accuracy (coarse mesh) solutions can potentially be used more effectively in an optimization.

Next Phase

Work in the next 3 months will include:

- Incorporation of feedback on the optimization system from beta users into both the user interface and algorithm development.
- Work on space-mapping or related methods for utilizing models of varying levels of approximation within an optimization.
- Study of collector optimization strategies using the software with individual(s) from industry.

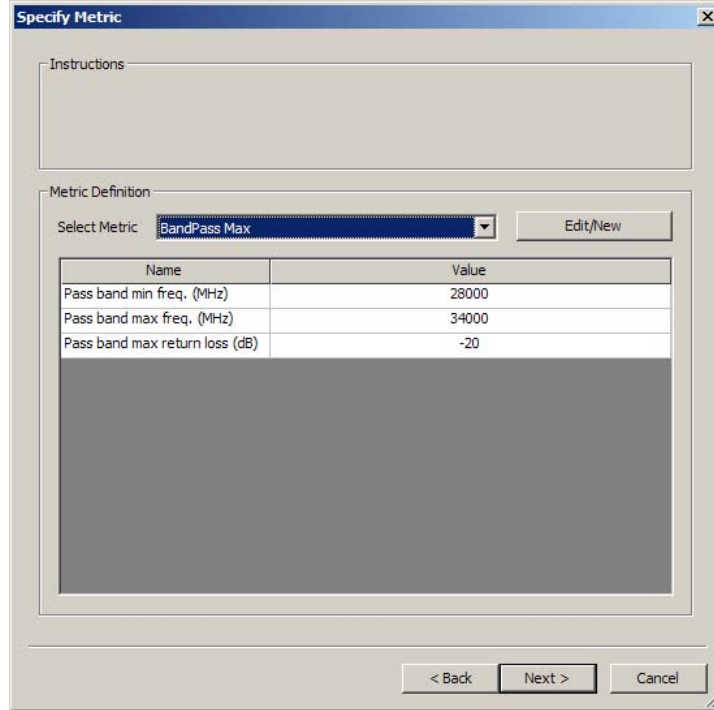


Fig. 1. Optimization panel used to select goal function and set any associated parameters. User picks the goal function to be used in the optimization. All goal functions are Python scripts that can have an arbitrary number of parameters. We have created a set of standard goal functions, and users can add their own functions as needed.

```
def Evaluate(params, case, metricparam):
    try:
        TargetCurrentParam = params.GetParameterByName(g_sTargetCurrent)
        dTargetCurrent = star.PySDoubleParameter(TargetCurrentParam).GetValue()
        db = star.PyORDoc(case.GetPostProcessor()).GetDatabase()

        i0 = db.GetIteration(db.GetNumIterations()-1)
        mat = star.PyORQMatrixObject(i0.GetQuantityResult(3))
        matv = mat.AsArrayObject()
        nNumRows = mat.GetNumRows()
        print matv[nNumRows -1][1]
        fMetric = (dTargetCurrent-matv[nNumRows -1][1])/dTargetCurrent
        metricparam.SetValue(fMetric)

    except:
        metricparam.SetValue(1000000)

    return
```

Fig. 2. Example goal function script. This script computes the difference between the gun current output by MICHELLE and a target value. User can create arbitrarily complicated metric functions in Python and then install them into Analyst for use in optimization.

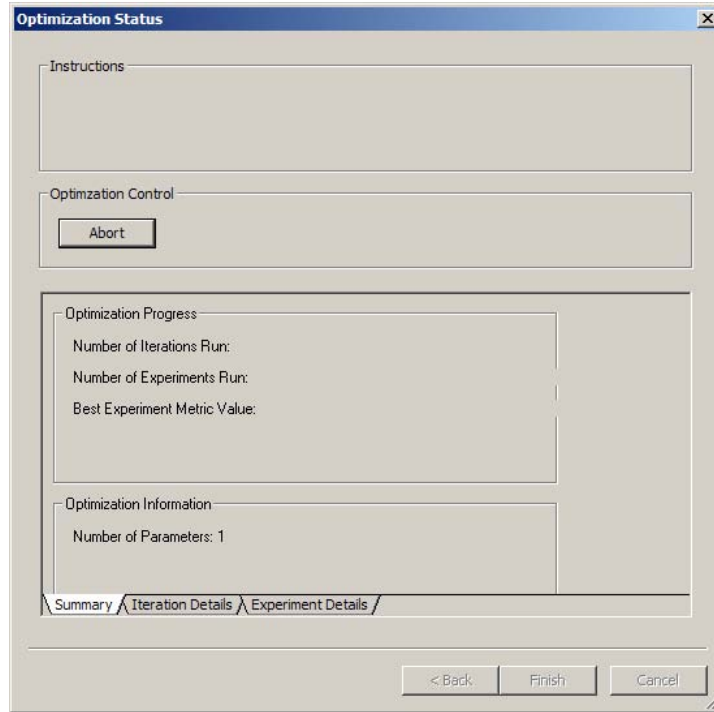


Fig. 3. This panel is displayed while the optimization is in progress. Results for each iteration, and for each individual experiment (analysis) can be found on the corresponding tabs. The optimization may also be aborted from this panel if necessary.