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
Base Plan Final Report

Contract/Purchase Order Number: N00014-05-C-0375
Simulation Technology & Applied Research Report Number: 06-SBIR-ONR-T4

Prepared for the Office of Naval Research

June 15, 2007

John F. DeFord, Ben Held, and Adam Bruss
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

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
15-06-2007

2. REPORT TYPE
Final

3. DATES COVERED *(From - To)*
Nov., 2005 - Apr., 2007

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, A. Bruss
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-T4

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

12. DISTRIBUTION AVAILABILITY STATEMENT
Unlimited/Unclassified

13. SUPPLEMENTARY NOTES

14. ABSTRACT
We have completed implementation of a variety of numerical optimization methods, including multi-directional search (MDS), differential evolution (DE), the Powell conjugate gradient method, and a zero-crossing technique. These methods have been extensively tested on several guns and collectors. The user interface to these techniques in Analyst has been completed and is included in Analyst Version 10. We have also completed development of a system for computing the manufacturing sensitivity of a design point. To support collector optimization we have recently created capabilities for the Analyst magnetostatics solver (MS3P) to output magnetic field files in a format that MICHELLE can read, and we have worked with SAIC on modifications to MICHELLE to use these files. This report was developed under a SBIR contract award for Solicitation topic N04-113.

15. SUBJECT TERMS
optimization,LINUX cluster,differential evolution,multi-directional search,Powell method,sensitivity,MICHELLE

16. SECURITY CLASSIFICATION OF:

a. REPORT	b. ABSTRACT	c. THIS PAGE
UU	UU	UU

17. LIMITATION OF ABSTRACT
UU

18. NUMBER OF PAGES
25

19a. NAME OF RESPONSIBLE PERSON
John F. DeFord

19b. TELEPONE 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 lest 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 al 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.

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

June 15, 2007

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

Dear Sirs and Ms:

Enclosed please find Final Report for the Base Plan of our Phase 2 SBIR project entitled "Computer-Aided Design and Optimization of High Performance Vacuum Electronic Devices", contract number N00014-05-C-0375. 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

Abstract

We have completed implementation of a variety of numerical optimization methods, including multi-directional search (MDS), differential evolution (DE), the Powell conjugate gradient method, and a zero-crossing technique. These methods have been extensively tested on several guns and collectors.

The user interface to these techniques in Analyst has been completed and is included in Analyst Version 10. We have also completed development of a system for computing the manufacturing sensitivity of a design point.

To support collector optimization we have recently created capabilities for the Analyst magnetostatics solver (MS3P) to output magnetic field files in a format that MICHELLE can read, and we have worked with SAIC on modifications to MICHELLE to use these files.

In February 2007 we were granted an extension of the period-of-performance of the base program until the end of April 2007. Through April 30, 2007 we have spent all of the funds in the base plan ($449k).

Executive Summary

The following items have been accomplished during this project and are currently available in the latest release of Analyst (Version 10).

- "Wizard-like" optimization user-interface extensions in Analyst for configuring and monitoring an optimization. Results of an optimization are presented real-time in spreadsheet-like controls and x-y plots.
- Architecture support in Analyst for optimization including caching of parameter vectors and associated metric values, scriptable constraint and metric functions, etc. Additionally, all optimizations can be asynchronously aborted and restarted if desired.
- Implementations of multidirectional search (MDS)[1], differential evolution (DE[2]), and Powell conjugate gradient[3] methods have been completed.
- Support for sensitivity analysis has been added to Analyst. This can be performed from any design point, but would typically be done on the result of an optimization.
- An extensive round of testing on all of the optimization algorithms has been completed.
- The Analyst user interface to MICHELLE is sufficiently complete to allow gun and collector optimizations, including support for external magnetic field calculation using the MS3P solver in Analyst.
- Validation via several gun and collector optimizations.
- Generation of full documentation of the Analyst/MICHELLE support including several self-contained examples for guns and collectors.
- A MICHELLE/Analyst training session was held in April 2007 introducing this technology to a select group of users. Industrial participants included CPII, L-3 ETI, L-3 ED, Northrop Grumman, Teledyne, and Beam Wave Research.

All of the tasks outlined for the base program in the original proposal have been completed except for the Analyst/CHRISTINE interface which was removed from the project by the program manager in exchange for a broadening the application of the optimization work on guns and collectors. These activities are discussed in more detail in the following sections.

Optimization User Interface

The user interface to the optimization capabilities takes the form of a "wizard", which is a set of panels that are navigated via "Next" and "Back" buttons. This format allows control over the setup process, walking the user through the following steps:

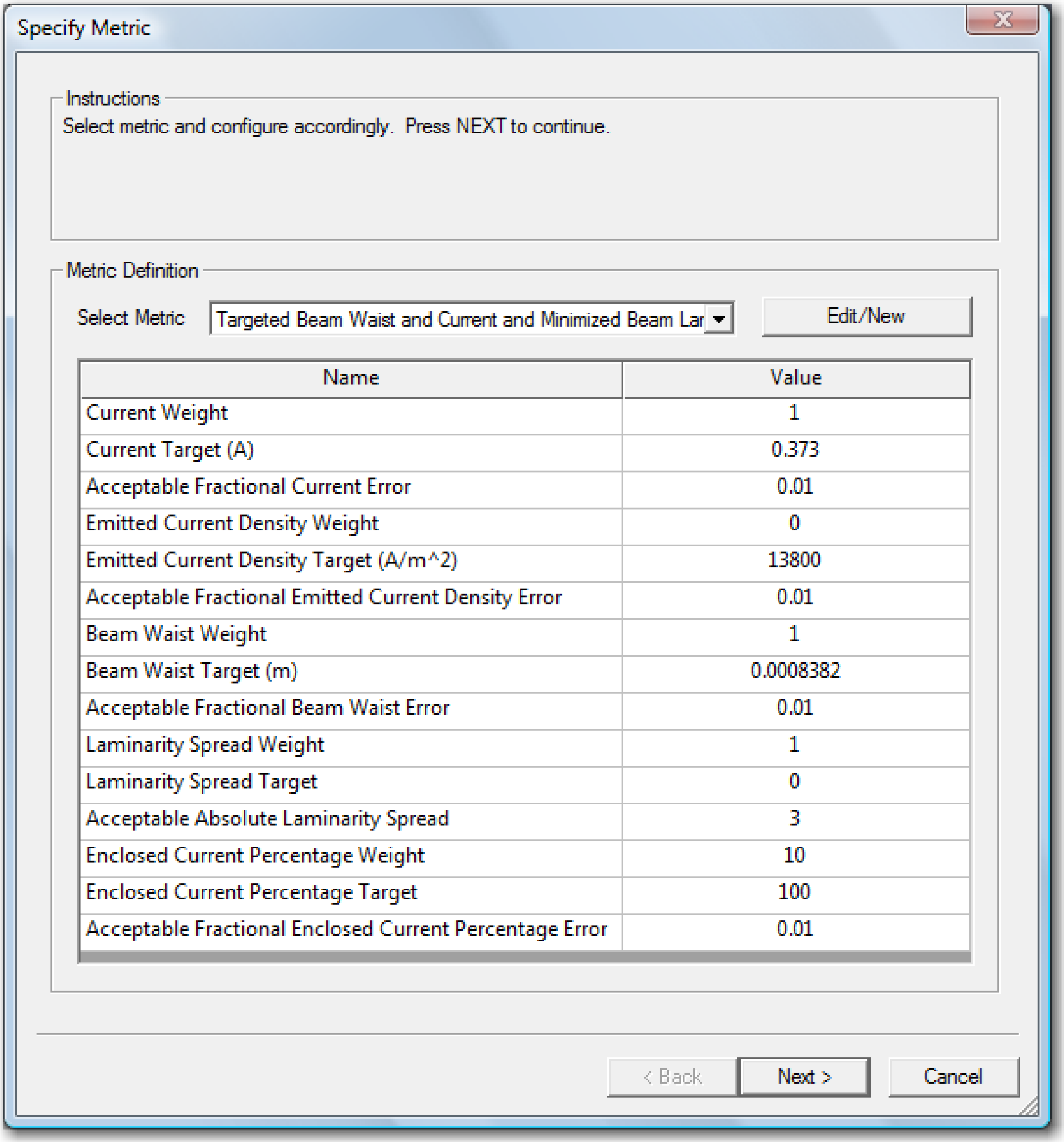
1. *Metric Specification*. A metric is represented by a Python module that takes a result database and list of geometric parameters and returns a real number representing the

[1] V. Torczon, *Multi-Directional Search: A Direct Search Algorithm for Parallel Machines*, Ph. D. thesis, Rice University, Houston, TX, May, 1989.

[2] R. Storn and K. Price, "Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim*., **11**, 1997, pp. 341-359.

[3] M. J. D. Powell, "An efficient method for finding the minimum of a function of several variables without calculating derivatives," *Compute J*., **7**, pp. 155-162, July 1964.

quality of the design. Several predefined functions are included, and the user can also define their own. A metric may contain configuration parameters such as target values, etc., and these parameters can be modified in this step.



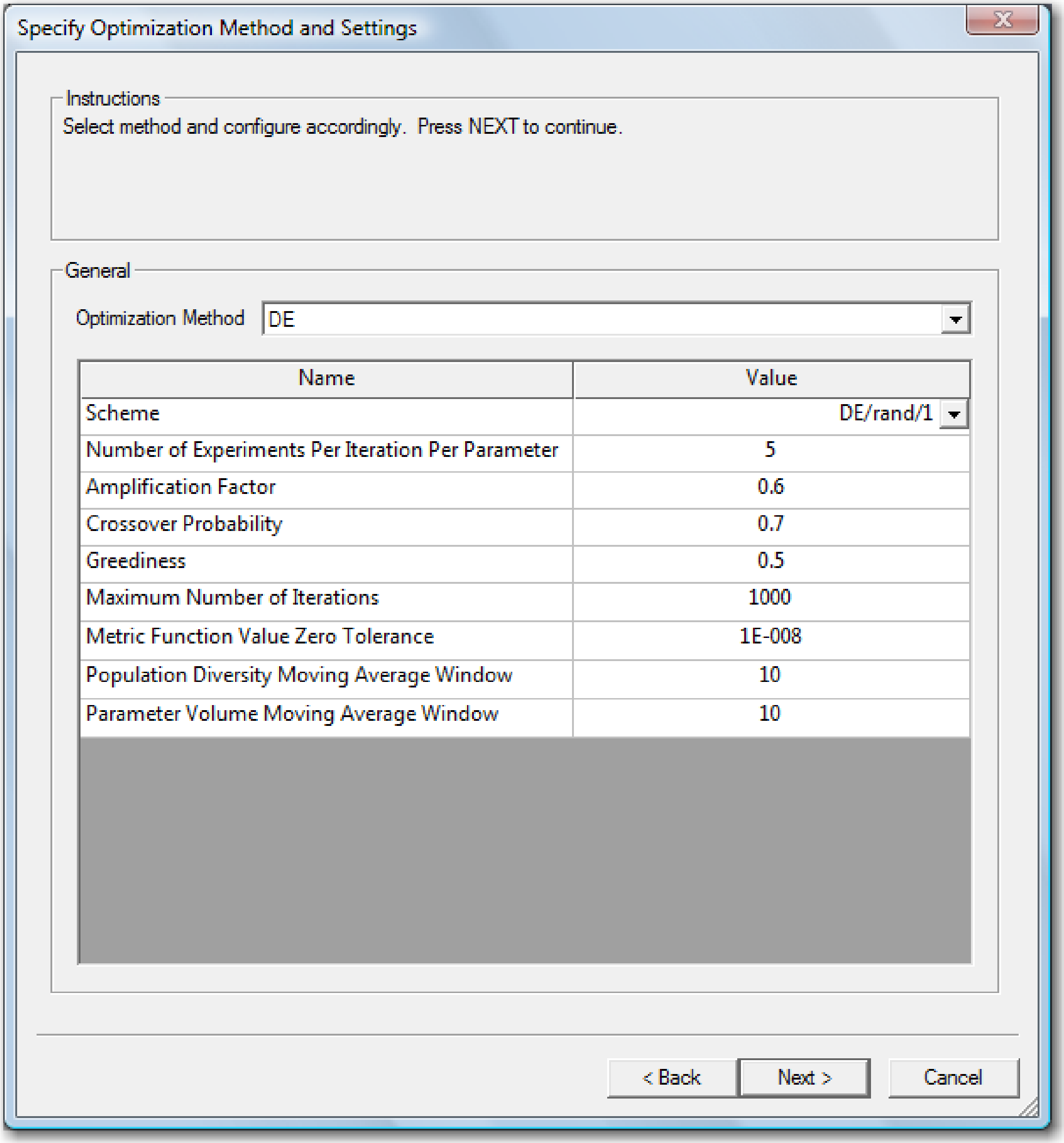
Select metric and configure accordingly. Press NEXT to continue.

Select Metric: Targeted Beam Waist and Current and Minimized Beam Lar

Name	Value
Current Weight	1
Current Target (A)	0.373
Acceptable Fractional Current Error	0.01
Emitted Current Density Weight	0
Emitted Current Density Target (A/m^2)	13800
Acceptable Fractional Emitted Current Density Error	0.01
Beam Waist Weight	1
Beam Waist Target (m)	0.0008382
Acceptable Fractional Beam Waist Error	0.01
Laminarity Spread Weight	1
Laminarity Spread Target	0
Acceptable Absolute Laminarity Spread	3
Enclosed Current Percentage Weight	10
Enclosed Current Percentage Target	100
Acceptable Fractional Enclosed Current Percentage Error	0.01

Fig. 1. The metric specification step in the optimization wizard.

2. *Method Specification.* Selection of the method (algorithm), and definition of values that control the method, e.g., the maximum number of analyses, stopping criteria, etc.



Specify Optimization Method and Settings

Instructions
Select method and configure accordingly. Press NEXT to continue.

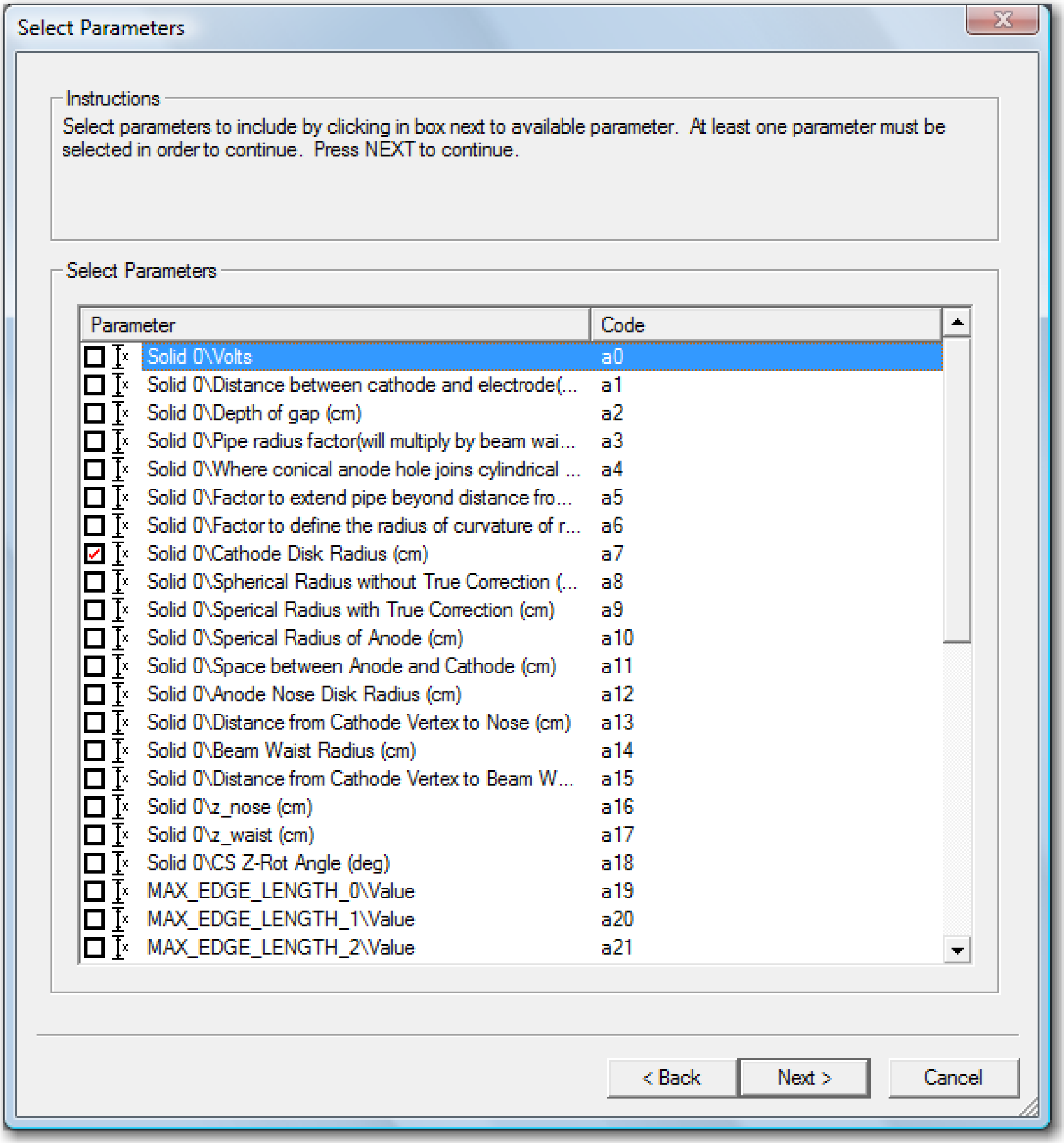
General

Optimization Method: DE

Name	Value
Scheme	DE/rand/1
Number of Experiments Per Iteration Per Parameter	5
Amplification Factor	0.6
Crossover Probability	0.7
Greediness	0.5
Maximum Number of Iterations	1000
Metric Function Value Zero Tolerance	1E-008
Population Diversity Moving Average Window	10
Parameter Volume Moving Average Window	10

Fig. 2. The method specification step in the optimization wizard.

3. *Parameter Selection*. The user is presented with a list of all parameters of the model (these include geometric, attribute/material, solver, and mesher parameters). Parameters are included via a checkbox next to the name, and the user can also modify a short code for each parameter that is used in the constraint definition.

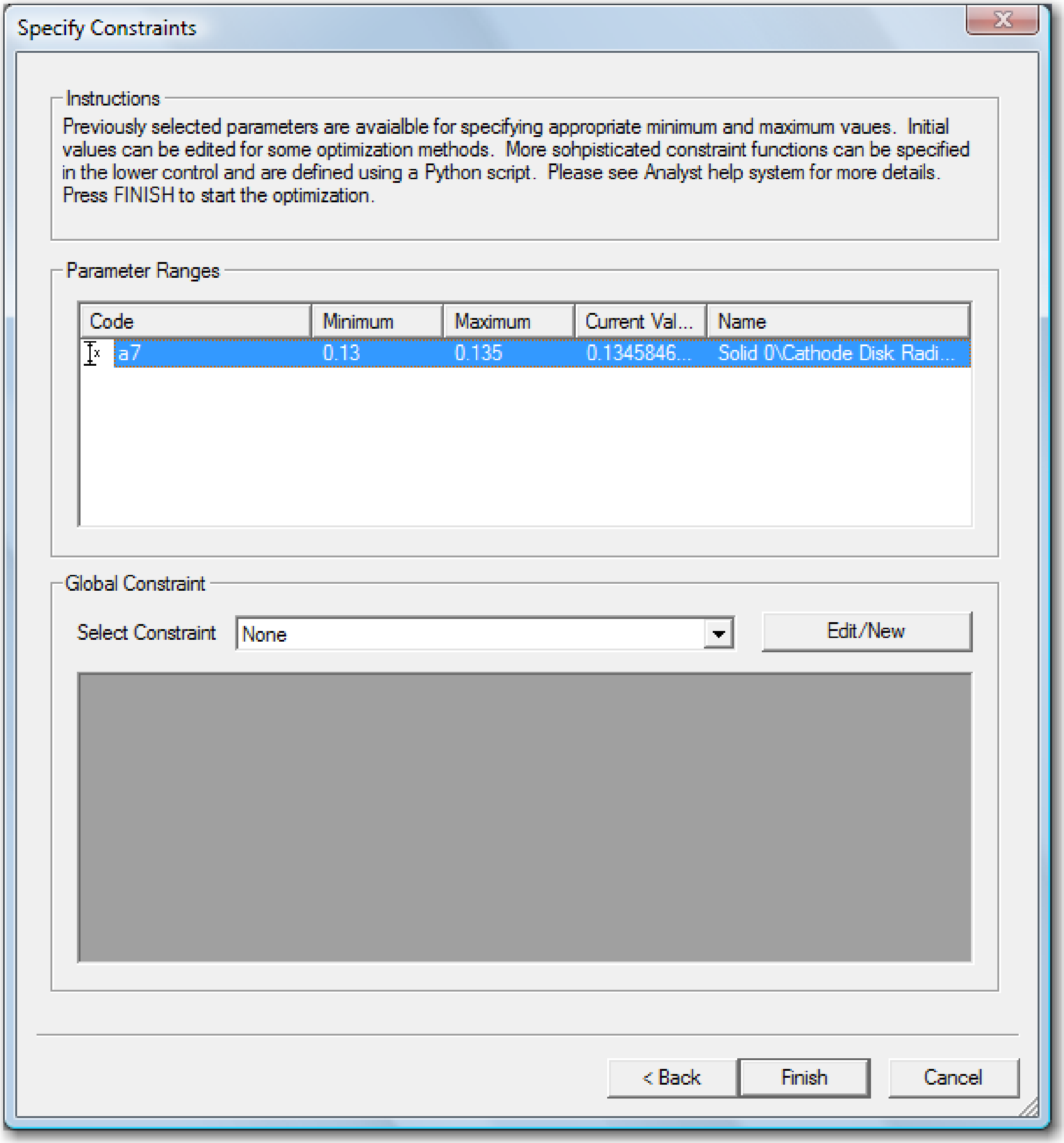


Select parameters to include by clicking in box next to available parameter. At least one parameter must be selected in order to continue. Press NEXT to continue.

	Parameter	Code
□	Solid 0\Volts	a0
□	Solid 0\Distance between cathode and electrode(...	a1
□	Solid 0\Depth of gap (cm)	a2
□	Solid 0\Pipe radius factor(will multiply by beam wai...	a3
□	Solid 0\Where conical anode hole joins cylindrical ...	a4
□	Solid 0\Factor to extend pipe beyond distance fro...	a5
□	Solid 0\Factor to define the radius of curvature of r...	a6
☑	Solid 0\Cathode Disk Radius (cm)	a7
□	Solid 0\Spherical Radius without True Correction (...	a8
□	Solid 0\Sperical Radius with True Correction (cm)	a9
□	Solid 0\Sperical Radius of Anode (cm)	a10
□	Solid 0\Space between Anode and Cathode (cm)	a11
□	Solid 0\Anode Nose Disk Radius (cm)	a12
□	Solid 0\Distance from Cathode Vertex to Nose (cm)	a13
□	Solid 0\Beam Waist Radius (cm)	a14
□	Solid 0\Distance from Cathode Vertex to Beam W...	a15
□	Solid 0\z_nose (cm)	a16
□	Solid 0\z_waist (cm)	a17
□	Solid 0\CS Z-Rot Angle (deg)	a18
□	MAX_EDGE_LENGTH_0\Value	a19
□	MAX_EDGE_LENGTH_1\Value	a20
□	MAX_EDGE_LENGTH_2\Value	a21

Fig. 3. The parameter selection step in the optimization wizard.

4. *Constraint Definition*. This includes defining both a range of validity for each selected parameter, and also an optional global constraint (via a Python function) that further restricts the domain.



Specify Constraints

Instructions

Previously selected parameters are avaialble for specifying appropriate minimum and maximum vaues. Initial values can be edited for some optimization methods. More sohpisticated constraint functions can be specified in the lower control and are defined using a Python script. Please see Analyst help system for more details. Press FINISH to start the optimization.

Parameter Ranges

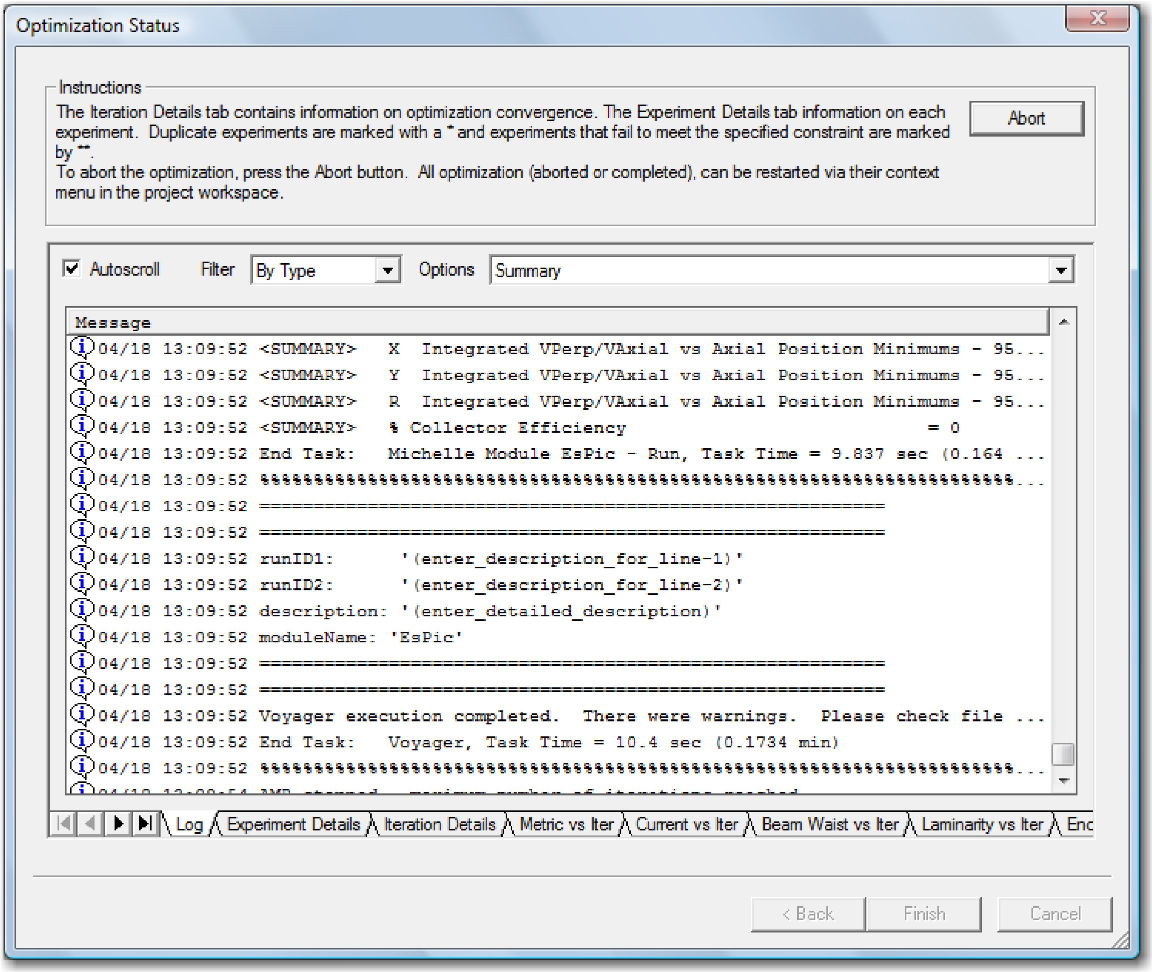
Code	Minimum	Maximum	Current Val...	Name
a7	0.13	0.135	0.1345846...	Solid 0\Cathode Disk Radi...

Global Constraint

Select Constraint: None

Fig. 4. The constraint definition step in the optimization wizard.

5. *Execution of Optimization*. Progress information is displayed in several different formats. A log tab shows the progress of every analysis (Fig. 5). The best case is retained, while all other cases are deleted during the optimization to avoid accumulation of unnecessary files and data. Moreover, the complete history of the experiments and resulting metric values can be copied from the Experiment Details table (Fig. 6) and pasted into an Excel spreadsheet for review or further manipulation (such as curve-fitting or other analysis). There are also XY plots of data such as metric progression versus iteration, etc. The data shown in these tables and plots can be fully customized via an optional specification in the metric script (this is covered in a later section). There is also an option for aborting the analysis.



Instructions

The Iteration Details tab contains information on optimization convergence. The Experiment Details tab infomation on each experiment. Duplicate experiments are marked with a * and experiments that fail to meet the specified constraint are marked by **.
To abort the optimization, press the Abort button. All optimization (aborted or completed), can be restarted via their context menu in the project workspace.

☑ Autoscroll Filter By Type Options Summary

```
Message
04/18 13:09:52 <SUMMARY>   X  Integrated VPerp/VAxial vs Axial Position Minimums - 95...
04/18 13:09:52 <SUMMARY>   Y  Integrated VPerp/VAxial vs Axial Position Minimums - 95...
04/18 13:09:52 <SUMMARY>   R  Integrated VPerp/VAxial vs Axial Position Minimums - 95...
04/18 13:09:52 <SUMMARY>   % Collector Efficiency                         = 0
04/18 13:09:52 End Task:   Michelle Module EsPic - Run, Task Time = 9.837 sec (0.164 ...
04/18 13:09:52 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%...
04/18 13:09:52 ==========================================================
04/18 13:09:52 ==========================================================
04/18 13:09:52 runID1:      '(enter_description_for_line-1)'
04/18 13:09:52 runID2:      '(enter_description_for_line-2)'
04/18 13:09:52 description: '(enter_detailed_description)'
04/18 13:09:52 moduleName: 'EsPic'
04/18 13:09:52 ==========================================================
04/18 13:09:52 ==========================================================
04/18 13:09:52 Voyager execution completed.  There were warnings.  Please check file ...
04/18 13:09:52 End Task:   Voyager, Task Time = 10.4 sec (0.1734 min)
04/18 13:09:52 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%...
04/18 13:09:54 AMR stopped - maximum number of iterations reached
```

Fig. 5. A log of the solver messages is shown in the optimization wizard.

Architecture

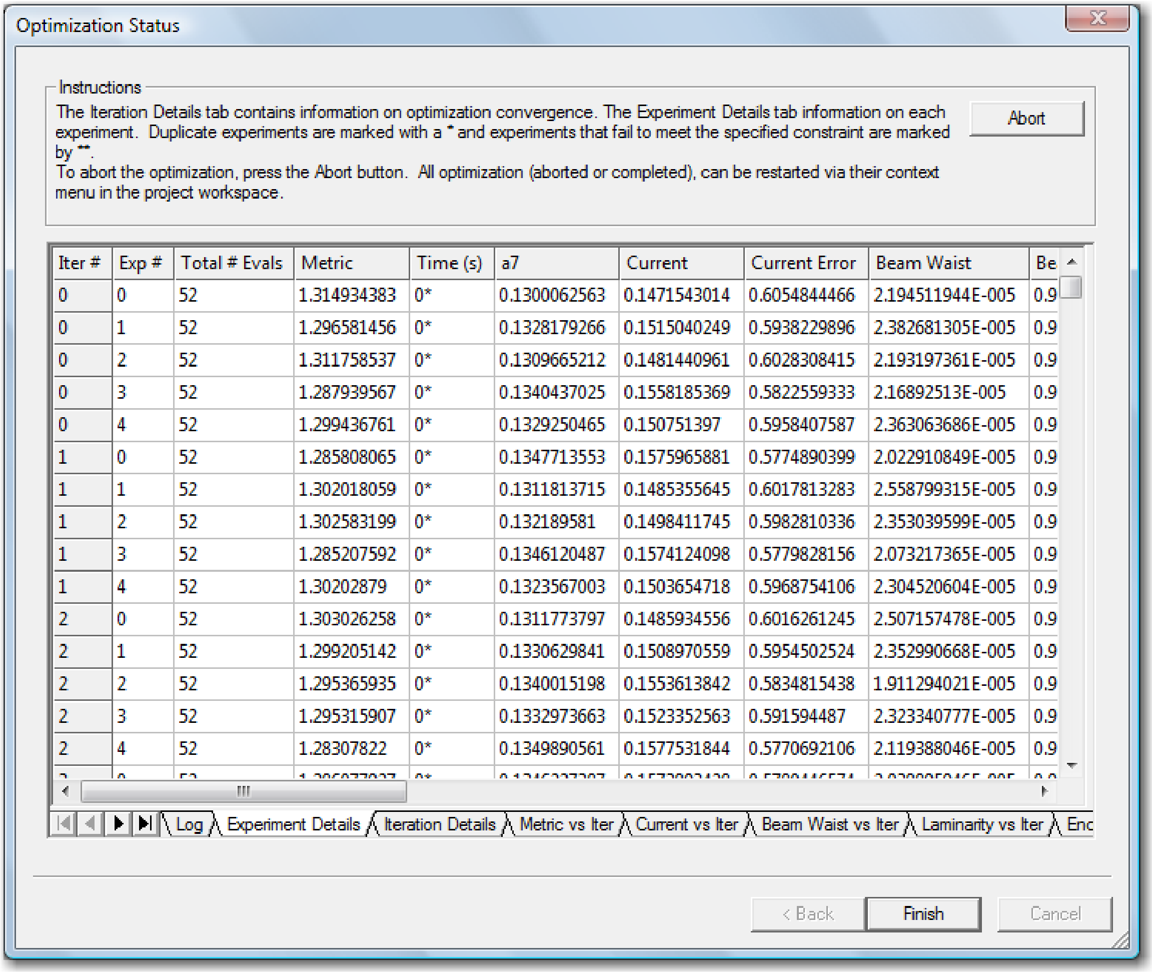
Caching

The optimization architecture in Analyst includes support for caching of parameter vectors and associated metric values. This cache is used to determine if an experiment (parameter vector) was previously analyzed. If so, its metric value is retrieved from the cache without the cost of an analysis.

This support is also utilized if an optimization is aborted and later restarted. If it is restarted without modification to the analysis or optimization settings (other than the stopping criteria), the optimization will continue from where it was aborted.

Constraints

In practice, all parameters associated with device optimization must, at a minimum, have a range of acceptable values. If an optimization method requests evaluation of an experiment that falls outside the valid parameter space, Analyst returns an infinite metric value and continues.



Instructions

The Iteration Details tab contains information on optimization convergence. The Experiment Details tab information on each experiment. Duplicate experiments are marked with a * and experiments that fail to meet the specified constraint are marked by **.
To abort the optimization, press the Abort button. All optimization (aborted or completed), can be restarted via their context menu in the project workspace.

Iter #	Exp #	Total # Evals	Metric	Time (s)	a7	Current	Current Error	Beam Waist	Be
0	0	52	1.314934383	0*	0.1300062563	0.1471543014	0.6054844466	2.194511944E-005	0.9
0	1	52	1.296581456	0*	0.1328179266	0.1515040249	0.5938229896	2.382681305E-005	0.9
0	2	52	1.311758537	0*	0.1309665212	0.1481440961	0.6028308415	2.193197361E-005	0.9
0	3	52	1.287939567	0*	0.1340437025	0.1558185369	0.5822559333	2.16892513E-005	0.9
0	4	52	1.299436761	0*	0.1329250465	0.150751397	0.5958407587	2.363063686E-005	0.9
1	0	52	1.285808065	0*	0.1347713553	0.1575965881	0.5774890399	2.022910849E-005	0.9
1	1	52	1.302018059	0*	0.1311813715	0.1485355645	0.6017813283	2.558799315E-005	0.9
1	2	52	1.302583199	0*	0.132189581	0.1498411745	0.5982810336	2.353039599E-005	0.9
1	3	52	1.285207592	0*	0.1346120487	0.1574124098	0.5779828156	2.073217365E-005	0.9
1	4	52	1.30202879	0*	0.1323567003	0.1503654718	0.5968754106	2.304520604E-005	0.9
2	0	52	1.303026258	0*	0.1311773797	0.1485934556	0.6016261245	2.507157478E-005	0.9
2	1	52	1.299205142	0*	0.1330629841	0.1508970559	0.5954502524	2.352990668E-005	0.9
2	2	52	1.295365935	0*	0.1340015198	0.1553613842	0.5834815438	1.911294021E-005	0.9
2	3	52	1.295315907	0*	0.1332973663	0.1523352563	0.591594487	2.323340777E-005	0.9
2	4	52	1.28307822	0*	0.1349890561	0.1577531844	0.5770692106	2.119388046E-005	0.9

Fig. 6. A table of experiments is shown in the optimization wizard.

Analyst offers two ways to define parameter constraints. The first and most commonly used is the specification of a minimum and maximum value for each parameter. These values can be set in one of the steps of the optimization wizard.

The second and more complex is a Python based global constraint. This functionally is useful if the parameter space is more difficult to define. For example, if you are varying the axial location of two adjacent plates in a collector, you need to make sure that they do not intersect each other. If you need to allow for large range in these locations, simply specifying a range for each parameter will not be sufficient, because much of that parameter space will cause the two plates to overlap. A global constraint script can be used instead to deal with this situation.

There are 3 functions that can be defined in a global constraint script:

- *Test()* - This function is called to determine if an experiment is allowed to be executed.
- *UpdateParameters()* - This function is called to update the metrics parameters.
- *ValueChanged()* - This function is called whenever a parameter value is changed. Its purpose is to allow conditional parameters. The main utility that this function

provides is a mechanism to control parameter visibility based on the value of specific parameter.

A template global constraint script is provided with Analyst and there is a simple user-interface that can be used to define and install new scripts.

Metrics

The metric (goal function) is used to assign a single value to the results of an experiment. Analyst is configured to always minimize the metric value and all metrics in Analyst are written as Python scripts. These scripts can identify parameters that are displayed to the user to specify weighting factors, stopping criteria, etc.

There are 5 functions that can be defined in a metric script:

- *IsCompatible()* - This function is called to determine if this metric is compatible with the model being optimized. This decision is typically based on the model's analysis type, but any accessible attribute of the model could be used to make this determination.
- *Execute()* - This function is called after a case is analyzed to compute the metric value.
- *UpdateParameters()* - This function is called to update the metrics parameters.
- *ValueChanged()* - This function is called whenever a parameter value is changed.
- *GetSupplementalDataInfo()* - This function is used to define additional custom data that the user is interested in seeing in tables/plots versus experiment and/or iteration.

A template metric script is provided with Analyst and there is a simple user-interface that can be used to define and install new scripts.

An important aspect of optimization of beam devices using the MICHELLE code is the creation of metric functions that capture the relevant characteristics of the beam. To this end we have created metrics for specific classes of design problems currently including guns and collectors. The one suitable for use in gun optimization includes terms related to the following beam quantities:

- *Beam Waist* for various current enclosed fractions:
  - Beam waist radius (or x & y transverse dimensions for sheet beams)
  - Associated axial positions
  - Current enclosed at beam waist (beam interception before the beam waist will not have the full current enclosed fraction)
- *Laminarity:*
  - Various quantities based on $v_\perp / v_\parallel$ as metrics for laminarity:
    - Mainly used for gun design without B-fields.
  - Metric for beam scalloping amplitude:
    - For gun design in the presence of B-fields.
- *Total Current:*
  - Metric for understanding where the beam interception occurs.
- *Emission Current Density:*
  - Maximum and minimum values for the emitted current density.
- *Beam Ripple:*

- Ripple amplitude.
- Ripple average radius.

This metric function allows the user to specify overall weight, target value, and acceptable error for each of these terms. The weights are used to combine the terms into a single metric value. If the acceptable error is met for each of the terms in the metric, a metric of zero is returned and the optimization stops.

We have also developed a metric for optimization of collectors that involves maximizing the collection efficiency.

Optimization Methods

The final implementations of DE, MDS, and Powell are now complete. The implementations are composed of a set of C++ classes that are integrated into the Analyst package.

The optimization package is architected so that it may be used to control a generic optimization process. As such, method-specific functions are declared as virtual or pure virtual in base classes, and are overloaded as necessary in derived classes that are specialized to a particular method. This architecture will make it very easy to add other methods to the library, without having to modify core functionality like process control, previous result lookup, and interactions with the analysis package.

MDS Algorithm

This approach is a direct search simplex method that is closely related to the Nelder-Mead method[4] in which a non-degenerate simplex of dimension $n+1$ is updated (for an n-dimensional parameter vector) at each step. The volume enclosed by the simplex reduces until it encloses an extremum of the objective function.

A step in the process begins with stored values of the simplex vertices (parameter vectors) and the associated objective function values. The vertex with the best (minimum) value of the objective function is identified, and a set of n search directions is defined by the edges that connect the best vertex to the remaining vertices in the simplex. The length of each edge defines the length of the associated step in that direction, i.e., the new sample points are obtained by "reflecting" each vertex about the best vertex, with the connecting edge defining the reflection plane normal. The new sample points, together with the previous best vertex, form a new simplex that is accepted for the next iteration if at least one of its vertices has a better objective function value than does the previous best vertex. If the simplex is not accepted there are is a simple set of expansion and/or contraction steps (wherein the search directions are maintained but the step-size is changed) that are performed to find an acceptable new simplex. The process terminates when the simplex nodes, and the corresponding goal function values, become "close enough". More precisely, the two criteria are:

$$\max\left(F\left(\vec{X}_{best}\right)-F\left(\vec{X}_i\right)\right)\Big|_{i=1\ldots NES}\leq\delta$$

[4] Nelder, J. A. and Mead, R. "A Simplex Method for Function Minimization." *Comput. J.* **7**, 308-313, 1965.

$$\sqrt{\frac{\sum_{k=1}^{NES} \| \vec{X}_k - \vec{X}_{mean} \|^2}{NES}} \leq \delta$$

where δ is a user defined parameter, and NES is the number of experiments in one iteration.

One advantage this method has over Nelder-Mead and some other direct search approaches is that it is inherently parallel, because the processing and function evaluations associated with the different search directions are independent and can be performed simultaneously on different processors. This feature will be exploited later on in Option 1 of the project when we enable Analyst to distribute independent MICHELLE analyses on individual processors of a cluster.

Analyst exposes several parameters for MDS:

- Reflection Scale Factor - Must be greater than 0. Default values is 2.0.
- Contraction Scale Factor - Must be between 0 and 1. Default values is 0.5.
- Expansion Scale Factor - Must be greater than 1. Default values is 2.0

DE Algorithm

The Differential Evolution (DE) method comes from a class of algorithms based upon evolutionary principles. To start the process, an initial “population” of random vectors $\{\mathbf{p}_{k,0}\}$ is created that all satisfy the parameter constraints. At each iteration (called a “generation”) of the process, new vectors are obtained from the previous set using the following concepts:

- *Mutation*. A new vector is formed via a combination of existing vectors of the form

  $$\mathbf{v}_{i,G+1} = \mathbf{p}_{r_1,G} + \alpha\left(\mathbf{p}_{r_2,G} - \mathbf{p}_{r_3,G}\right)$$

  where *r1*, *r2*, and *r3* are randomly chosen integers.

- *Recombination (also called crossover)*. A candidate “child” vector is formed by taking some (randomly selected) parameter values directly from the parent $\mathbf{p}_G$, and the rest from the differential combination vector $\mathbf{v}_G$ i.e.,

  $$\mathbf{u}_{i,G+1} = \begin{cases} \mathbf{v}_{i,G}\ , i \in S \\ \mathbf{p}_{i,G}\ , i \notin S \end{cases}$$

- *Selection*. A parent vector is replaced with a child vector if the objective function is reduced. Otherwise, additional children are created and tested until either one is found that reduces the objective function or some maximum number of offspring is reached. If no child is more “fit” than the parent, the parent passes to the new generation (if they are not eliminated by the aging criterion below).

- *Aging*. A vector can only “survive” for a limited number of generations, regardless of its “fitness”.

In addition to the basic DE algorithm (named DE/rand/1), we implemented several different schemes that determine how the subsequent generation is constructed (*r1*, *r2*, *r3*, and *r4* are randomly chosen integers).

Table 1. DE schemes and expressions for subsequent generation construction.

Scheme Name	Expression
DE/best/1	$\mathbf{v}_{i,G+1} = \mathbf{p}_{best,G} + \alpha\left(\mathbf{p}_{r_1,G} - \mathbf{p}_{r_2,G}\right)$
DE/best/2	$\mathbf{v}_{i,G+1} = \mathbf{p}_{best,G} + \alpha\left(\mathbf{p}_{r_1,G} + \mathbf{p}_{r_2,G} - \mathbf{p}_{r_3,G} - \mathbf{p}_{r_4,G}\right)$
DE/rand-to-best/1	$\mathbf{v}_{i,G+1} = \mathbf{p}_{i,G} + \lambda\left(\mathbf{p}_{best,G} - \mathbf{p}_{i,G}\right) + \alpha\left(\mathbf{p}_{r_2,G} - \mathbf{p}_{r_3,G}\right)$

Analyst exposes several parameters for DE:

- *Scheme* – see Table 1.
- Number of Experiments Per Iteration Per Parameter - determines the size of the population of a generation. It is expressed in terms of the size of the population per parameter (dimension). So if you are optimizing over 2 parameters, and this value is 5, there will be 10 experiments per iteration. Making this number too small (< 3) can significantly reduce the number of potential trial experiments and therefore increase the likelihood of stagnation.
- *Amplification Factor* - α in the above table. This value determines how many of the potential trial experiments differ from each other. This number is typically between 0.0 and 2.0. It is not recommended to set this value to exactly 1.0 as this reduces the number of different potential experiments. A larger amplification factor increases the probability for escaping a local optimum, but a value of > 1.0 can cause the convergence speed to decrease. If one suspects a local optimum has been found, the amplification factor should be increased.
- *Crossover Probability* - use in the recombination portion of the algorithm. Typical values are between 0.0 and 1.0, but it is not recommended to set it to 1.0 exactly as the risk of permanent stagnation is increased. A larger crossover probability often speeds up convergence.
- *Greediness* - λ in the above table. Only used for the DE/rand-to-best/1 scheme.

Powell Conjugate Gradient Algorithm

DE is a reliable statistical method for use in design optimization, and it has the ability to escape from local minima in order to find a more optimal solution. However, DE is relatively expensive in terms of the number of metric function evaluations required to achieve a converged result. In cases where a local minimum will suffice, more efficient methods are available that rely on local derivative information or on the ability to find minima along specified lines through parameter space.

Steepest descent methods are commonly used to find local minima, but they have the disadvantage that they can be very slow to converge in some cases. An example that illustrates this is one that has two parameters and the desired optimum lies along a long narrow "valley" in the metric function value. In this situation steepest descent tends to oscillate across the valley instead of moving down the valley.

A method for overcoming this limitation was developed by Powell, in which subsequent search directions are orthogonal to each other and the optimization proceeds by line minimization in the new direction starting from the best point from the previous iteration. Searching along "conjugate" directions dramatically improves the performance of descent algorithms in many cases, and we have implemented the Powell method for use in the Analyst optimization library. Preliminary tests show good performance on simple gun problems, with total number of analyses substantially reduced as compared with DE and MDS. We will be doing more testing on this method over the next few weeks.

The Powell method is an algorithm that searches for a local minimum of an objective function for a set of linearly independent direction vectors without knowledge of the derivatives. It is one of several algorithms classified at conjugate direction methods.

The implementation in Analyst is based on the description found in *Numerical Recipes in C*[5]. It makes use of Brent's[6] method for line minimization.

Sensitivity (Manufacturing Tolerance)

Optimal design points must necessarily be manufacturable to be useful, and extreme sensitivity to parameter values usually leads to production problems. In order to determine the sensitivity of a design we have implemented an analysis sequence that uses solutions in the neighborhood of the design point to estimate metric derivatives. The metric function is expanded in a Taylor series about the design point using the first and second order finite-difference derivatives, and the series is examined to determine the parameter space that keeps the metric within an acceptable degradation from its value at the design point.

The sensitivity calculation is accessed via the optimization panels, and is treated as another kind of optimization sequence in the sense that it is selected and configured on the same panel that is used for optimization procedures.

The sensitivity calculation has two setup parameters: (1) a parameter delta that is used to compute the finite-difference derivatives about the design point, and (2) the degradation in the metric function that can be tolerated. The analysis sequence will generally involve $2n$ calculations, where n is the number of parameters, and the results of these calculations are used to estimate the first and second order derivatives of the metric function about the design point. The $2n$ calculations are all independent, and thus may be done simultaneously on a cluster or parallel computer, and support for this approach will be implemented in the Option portion of the project.

To determine manufacturing sensitivity we compute the range of parameter values that keep the metric within acceptable bounds. The control parameters for this method are a maximum acceptable deviation of the metric value (ε) and a fraction of the range for each parameter (f). The fraction of range gets multiplied by the parameter range to get a Δ that is used to estimate local derivatives of the goal function near the design point.

Variables that are used in the following discussion are described in the following tables.

[5] William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, Numerical Recipes in C: The Art of Scientific Computing , 2nd edition, Cambridge Univ. Press, N.Y., 1992.
[6] Brent, R.P. 1973, *Algorithms for Minimization without Derivatives* (Englewood Cliffs, NJ: Prentice-Hall), Chapter 7.

Table 2. Known variables in the sensitivity calculation.

Name	Description
p_0	Parameter value at design point.
G_0	Metric value at design point.
p_{max}	Maximum parameter value.
p_{min}	Minimum parameter value.
f	Fraction of parameter range used to compute Δ.
ε	Maximum acceptable deviation of the metric value.

Table 3. Unknown variables in the sensitivity calculation.

Name	Description
Δp_+	Maximum acceptable deviations in parameter value to keep metric within ε.
Δp_-	Maximum acceptable deviation in negative direction in parameter value to keep metric within ε.

Table 4. Sensitivity calculation derived result quantities.

Name	Description
Δ	Used for estimation of local derivatives. $\Delta = f\,(p_{max}\text{-}p_{min})$
G_{mfg}	Maximum acceptable metric value. $G_{mfg} = G_0 + \varepsilon$.
D_1	First order derivative.
D_2	Second order derivative.
p_-	Minimum acceptable parameter value. $p_- = p_0 + \Delta p_-$
p_+	Maximum acceptable parameter value. $p_+ = p_0 + \Delta p_+$
R	Manufacturing tolerance range. $R = \lvert\Delta p_-\rvert + \lvert\Delta p_+\rvert$

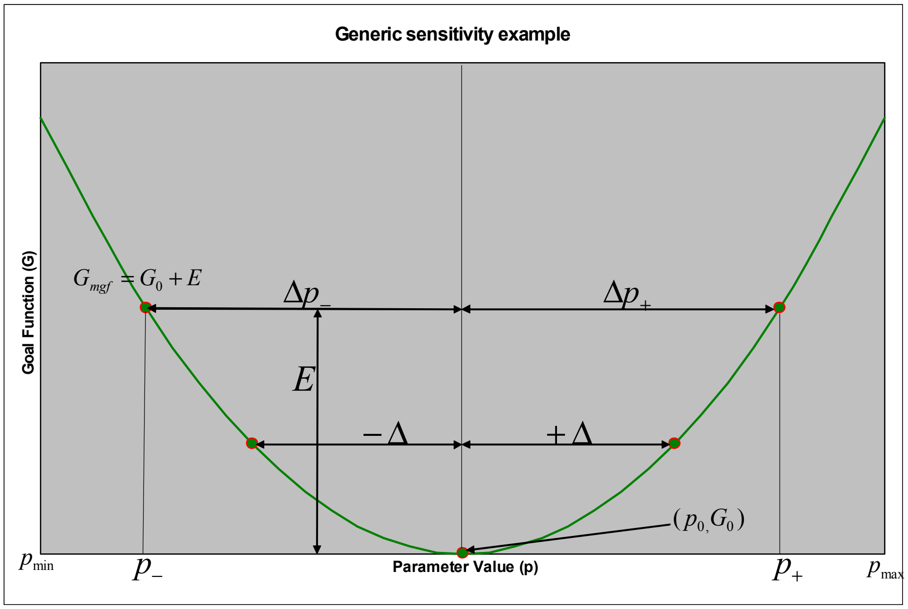


Fig. 7. Illustration of sensitivity calculation.

To compute the unknowns Δp_+ and Δp_- first compute the 1st and 2nd order derivatives (D_1, D_2) using the finite-difference approximation. They are obtained by solving the following quadratic equation:

$$\Delta p^2(\tfrac{1}{2}D_2) + \Delta p(D_1) + (-G_0\varepsilon) = 0$$

which has solutions of the form

$$\Delta p = \frac{D_1 \pm \sqrt{D_1^{\,2} - 4(\tfrac{1}{2}D_2)(-G_0\varepsilon)}}{D_2}$$

The two solutions to the quadratic equation are added on either side of the design point to form the manufacturing tolerance range. This indicates how accurate the manufacturing must be to ensure the metric value stays within an acceptable deviation from the design point.

Algorithm Testing

We have tested the algorithms on a variety of analytic functions, including:

1. Simple quadratic in two dimensions.
2. A discrete function of two parameters of the form:

   $$R[x,y] = \begin{cases} rand(0,1)_{i,j} \; -100 \le \left(i = \text{int}(x/dx)\right) \le 100, -100 \le \left(j = \text{int}(y/dy)\right) \le 100 \\ \qquad\qquad\qquad\qquad \infty \quad otherwise \end{cases}$$

   where $rand(0,1)_{i,j}$ is a random number in the range [0,1] that is generated for each *i,j* position on the grid.
3. The square root of *R* defined above.
4. The square of R defined above.
5. A sum of squares of thirty variables.

These functions were chosen to stress the algorithms and reveal any problems in the implementation, not because they are representative of real-world design optimization problems. Based on these tests we believe that all of the methods are now working properly.

Magnetic Field Support

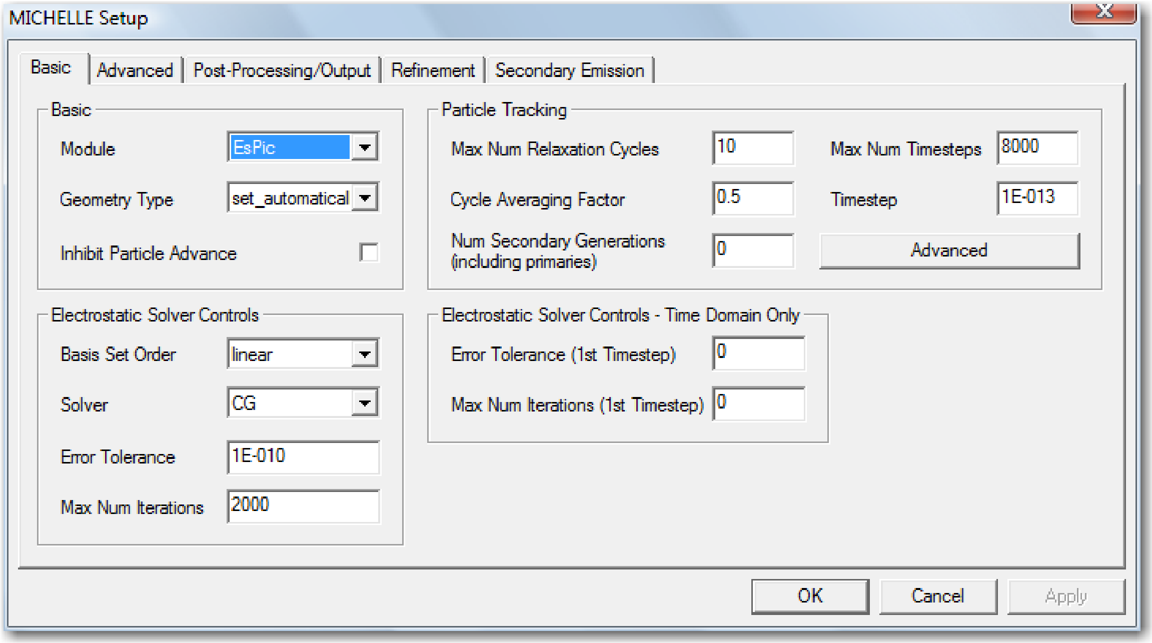
In anticipation of the need to manipulate external magnetic fields to optimize collector efficiency, we have added support to Analyst to export magnetic fields in a form that can be used by MICHELLE. Although MICHELLE previously supported external magnetic fields, the requirement that the fields be specified at the nodes of the MICHELLE mesh was too restrictive for simple use within the Analyst framework, primarily because the mesh for MICHELLE is not necessarily the mesh used in the magnetic field computation.

Analyst was extended to drive one of the MICHELLE input translators that allows for the specification of the magnetic field on a regular grid, from which it interpolates the fields at the nodes of the MICHELLE mesh. This allows the particle and magnetic field calculations to be completely independent of each other.

MICHELLE Interface in Analyst

The user interface to MICHELLE within Analyst is essentially complete. The setup panels for MICHELLE have been reworked several times since their original implementation, and have converged on a reasonable compromise between simplicity of presentation and ready access to most parameters.

The MICHELLE setup is covered by a tabbed dialog in which the majority of the parameters that are typically modified are covered in the first tab (Fig. 8). The remaining tabs contain more advanced options for magnetic fields, secondary controls, automatic mesh refinement settings, etc.



Basic

Module: EsPic
Geometry Type: set_automatical
Inhibit Particle Advance □

Particle Tracking

Max Num Relaxation Cycles: 10
Max Num Timesteps: 8000
Cycle Averaging Factor: 0.5
Timestep: 1E-013
Num Secondary Generations (including primaries): 0

Electrostatic Solver Controls

Basis Set Order: linear
Solver: CG
Error Tolerance: 1E-010
Max Num Iterations: 2000

Electrostatic Solver Controls - Time Domain Only

Error Tolerance (1st Timestep): 0
Max Num Iterations (1st Timestep): 0

Fig. 8. Basic setup tab for MICHELLE in Analyst.

Gun Optimization

Two guns were investigated during the base project. Only 2D structures were explored to minimize runtimes (support for 3D optimization is complete and has been tested).

The CPI XK-8050 gun was chosen as it is a well understood problem. We followed several design steps that an engineer would ordinarily use as they develop a new gun using the MICHELLE code. At each step, the Analyst optimizer was used to adjust geometry and other parameters to meet the objectives of that step. We used a variation of the XK-8050 gun design as a starting point (see Fig. 9).

The MICHELLE runs were configured to run for 55 relaxation cycles with a cycle-to-cycle averaging factor of 0.3. The geometry was subdivided into 3 regions for meshing purposes and a final mesh of approximately 80,000 elements (Fig. 10) was used throughout the optimizations resulting in an average analysis time of 90 seconds per run.

The initial pre-optimized case is shown in Fig. 11, where the current was about 40% too low and the beam radius was out of the range of specification. The four cases 1, 2, 3a &

3b are outlined below and they cover starting with the initial design and getting the right current, to adding a magnetic field and reducing the beam ripple in that field.

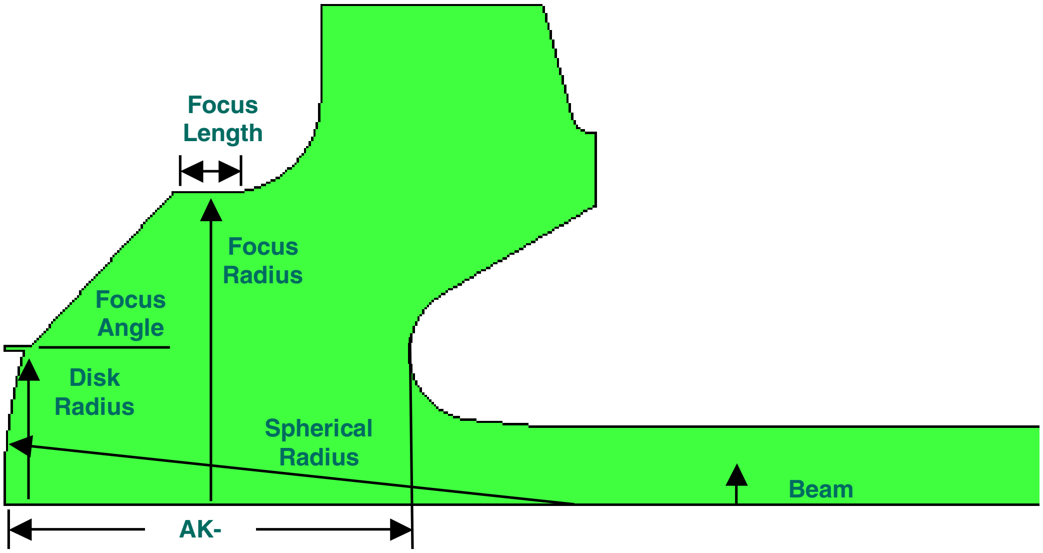


Fig. 9. CPI XK-8050 gun geometry. This shortened geometry was sufficient for optimizing the target current and target beam waist radius. A longer beam pipe was included when beam ripple was the object of the optimization.

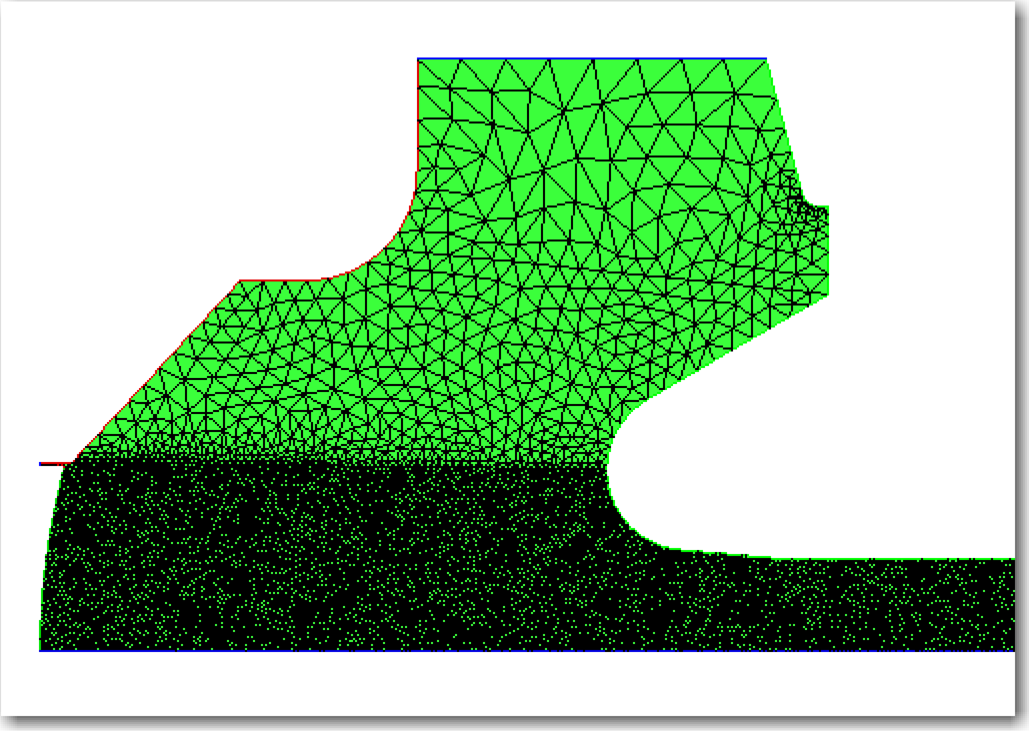


Fig. 10. A typical mesh of the XK-8050 gun consisting of ~80,000 elements.

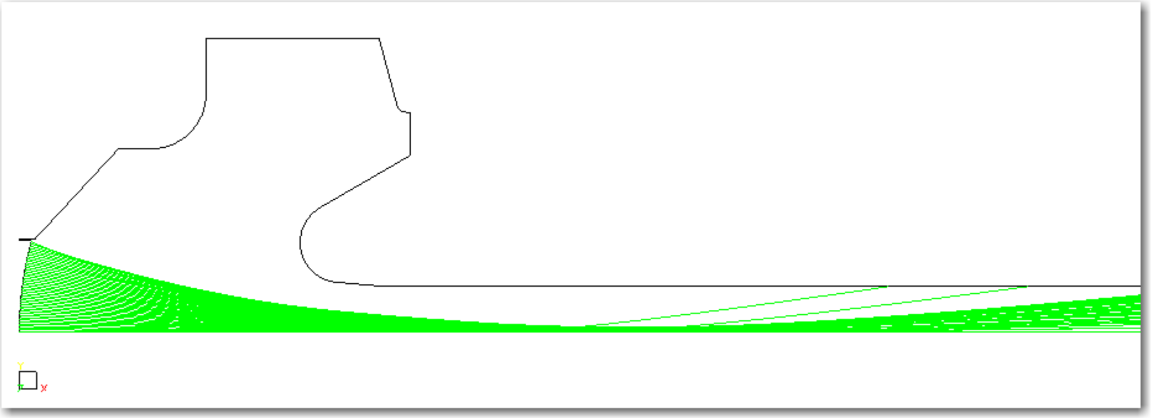


Fig. 11. Particle beam – initial design point. Here the beam current was about 75% of the target current and the beam waist was much smaller than the optimal value near 50% of the beam pipe radius.

Step 1: *With no applied B-field, vary the anode-cathode gap to produce a desired current.* The beam radius needed to be in some range, and that it fell into that range was a requirement to move forward to the next step. But it was not set as a specific constraint, although the designer may choose to do so. The results are shown in Table 5 and Fig. 12. With the initial gap spacing, the current was 0.60 of the desired current. After optimization using the Powell conjugate-gradient method, the gap spacing was 0.76 of the initial gap spacing to get a target current within 0.015% of the target current. It found this result within 17 MICHELLE runs.

Table 5. Results of XK-8050 gun optimization for step 1 using the Powell conjugate-gradient method. The anode-cathode gap was varied to obtain a specified current.

	Initial	Optimal
Normalized Anode-Cathode Gap	1.0	**0.7590452**
Actual Current/Target Current	0.596669	**1.00015**
Number of Analyses	**-**	**17**

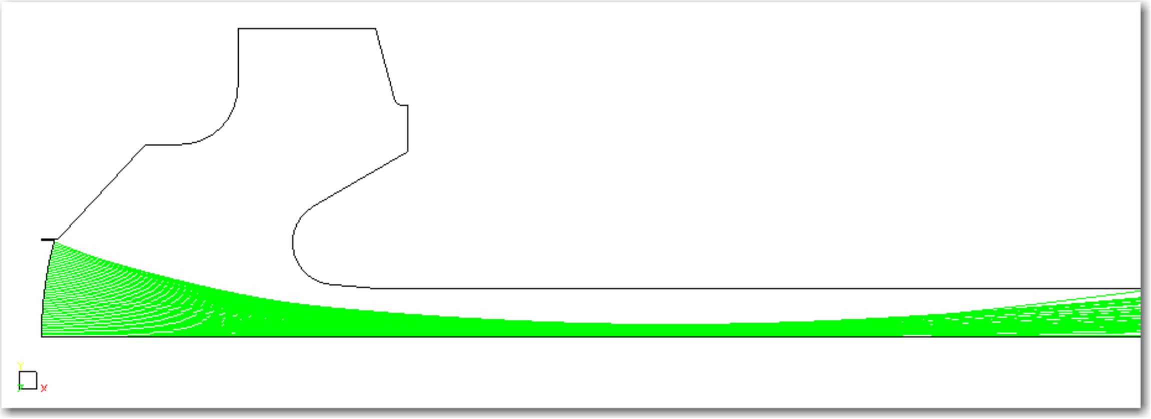


Fig. 12. Particle beam – design point resulting from step 1.

Step 2: *With no applied B-field, vary the focus electrode angle, radius, and length to improve the laminarity of the beam at the 95% current enclosed beam waist radius axial position while maintaining the current and target beam waist radius obtained in step 1.* As before, the beam radius needed to be in some range, that it fell into that range was a requirement to move forward to the next step, and it was not set as a specific constraint. The laminarity is an integrated measure of the beam's radial velocity over the axial velocity at the axial location of the beam minimum – chosen at the 95% current enclosed beam waist.

Laminarity for axisymmetric systems is defined with the following equation, where

l - laminarity at beam waist
i – particle index
w – macroparticle weight
v_r – radial velocity
v_z – axial velocity

$$l = \sum_{i_{0\%}}^{i_{95\%}} w_i \frac{v_r^i}{v_z^i}$$

The results are shown in Table 6 and Fig. 13. In this case, the laminarity was improved to 61.5% of its original value by modifying the focus electrodes while maintaining the current obtained in step 1. This result took 583 steps to achieve because the DE method was used. Powell was first used, but it fell into a local minimum and was not effective.

Table 6. Results of XK-8050 gun optimization for step 2. This optimization was done using the DE method as the Powell method immediately fell into a local minimum. The focus electrode angle, radius, and length were varied to improve beam laminarity while maintaining a specified current.

	Initial	Optimal
Normalized Focus Angle	1.0	**1.09303**
Normalized Focus Length	1.0	**0.546376**
Normalized Focus Radius	1.0	**1.371066**
Normalized Laminarity (95% current enclosed)	1.0	**0.615**
Actual Current/Target Current	1.00015	**1.000266**
Number of Analyses	**-**	**583**

Step 3: *Add in external magnetic field and reduce beam ripple while maintaining current and average beam radius by*

a. varying the focus electrode angle, radius, and length, or
b. varying the magnetic field location and strength.

Steps 1 & 2 are used by designers as a way to converge towards step 3. Step 3, with the addition of a magnetic field, is the final design step because it captures the beam and transports it to the interaction cavity.

In step 2, the designer made changes to maximize beam laminarity so that an applied magnetic field will have a better chance of transporting that beam with minimum beam ripple. In that case, the focus electrode was modified using the three geometric parameters described in Table 6. It is expected that a beam with high laminarity near its beam waist with no magnetic field will have low ripple when the magnetic field is applied. In Step 3a, in the presence of a solenoidal magnetic field, the same parameters are varied. However, here we also choose a final average beam radius.

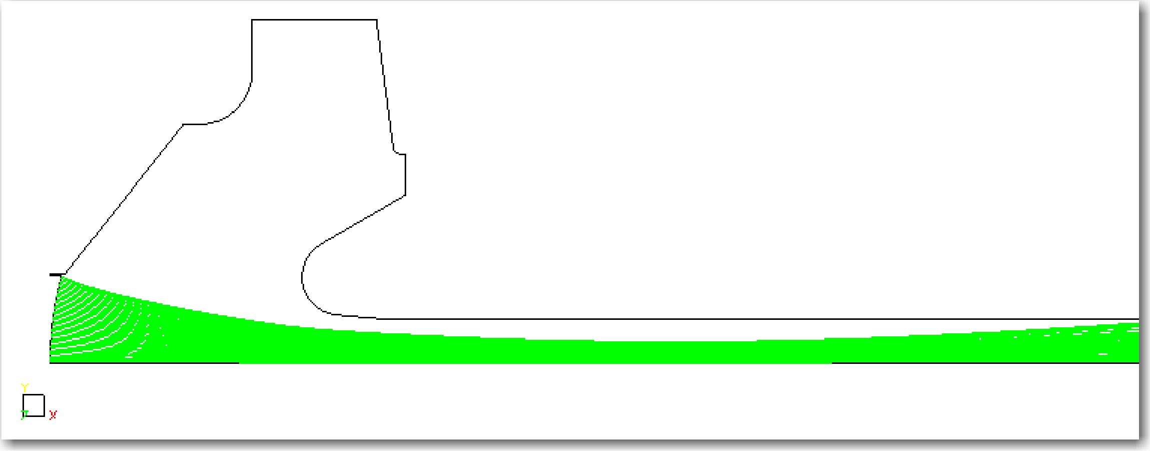


Fig. 13. Particle beam – design point resulting from step 2.

The results are shown in Table 7 and Fig. 14. The beam ripple is calculated by choosing the peaks and valleys of the beam after it has been focused and is fully inside the magnetic field. The 95% current–enclosed profile is used, and the value is calculated as ½ that peak value minus the valley value, and the average radius is the average of those same data. We have several ways to determine this. One way is by choosing the absolute peak and absolute minimum, another by averaging all peaks and averaging all minimums, and a third by using Fourier transforms of the data to resolve the ripple amplitude and average radius. The table shows that the ripple was reduced about 13%, and that the focus electrode geometric parameters were modified fairly substantially. Although the beam ripple was reduced, it was still noticeable.

Table 7. Results of XK-8050 gun optimization for step 3a using DE. An external magnetic field was introduced and the focus electrode angle, radius, and length were varied to reduce beam.

	Initial	Optimal
Normalized Focus Angle	1.09303	**0.98787**
Normalized Focus Length	0.546376	**0.982649**
Normalized Focus Radius	1.371066	**1.012328**
Normalized Beam Ripple	1.0	**0.87**
Actual Current/Target Current	1.000266	**0.9999**
Number of Analyses	**-**	**324**

Step 3b fixed the focus electrode at the geometry resulting from step 2, but added the magnetic field. However, in this case the magnetic field was allowed to vary in magnitude and also location along the axis. Otherwise, the field shape was not changed. Again, minimizing beam ripple was the goal, with the same average beam radius chosen.

The results are shown in Table 8 and Fig. 15. The figure shows that the beam ripple is almost unnoticeable. The table indicates that reducing the strength by 14% and shifting the field location 0.01759 in. closer to the cathode reduced the beam ripple to ¼ of its original value.

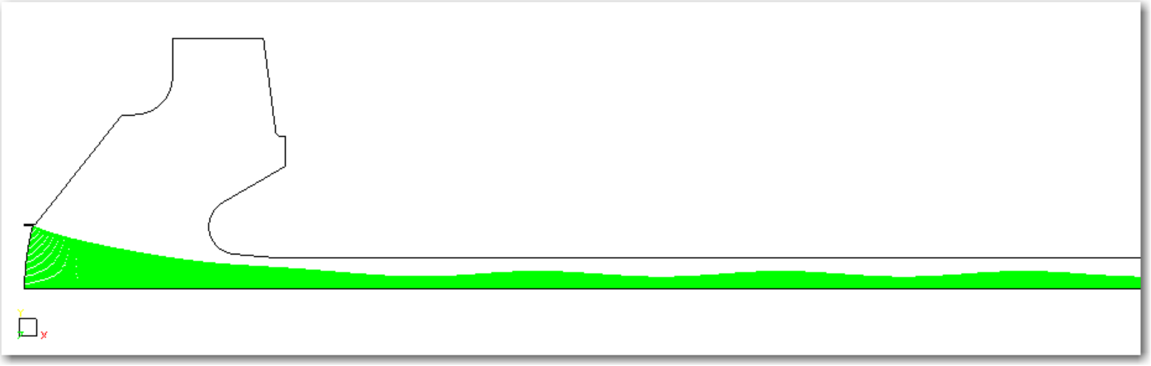


Fig. 14. Result of introducing magnetic field onto the result from step 2 (Fig. 13) and modifying the three focus element geometric parameters to achieve minimum beam ripple with a given average beam radius.

It is clear that Step 3b produced the most significant reduction in beam ripple (~75%). Fig. 16 shows the ripple of the initial geometry and the optimal geometry. It is important to note that the starting geometry was the result of an actual design. The optimization improved its performance.

Table 8. Results of XK-8050 gun optimization for step 3a using DE. An external magnetic field was introduced and the magnitude of the field was modified and the field was shifted along the axis to achieve minimum beam ripple with a given average beam radius.

	Initial	Optimal
Magnetic Field Strength	1.0	**0.8649**
Magnetic Field Offset (inches)	0.0	**-0.01759**
Normalized Beam Ripple	1.0	**0.247**
Number of Analyses	**-**	**123**

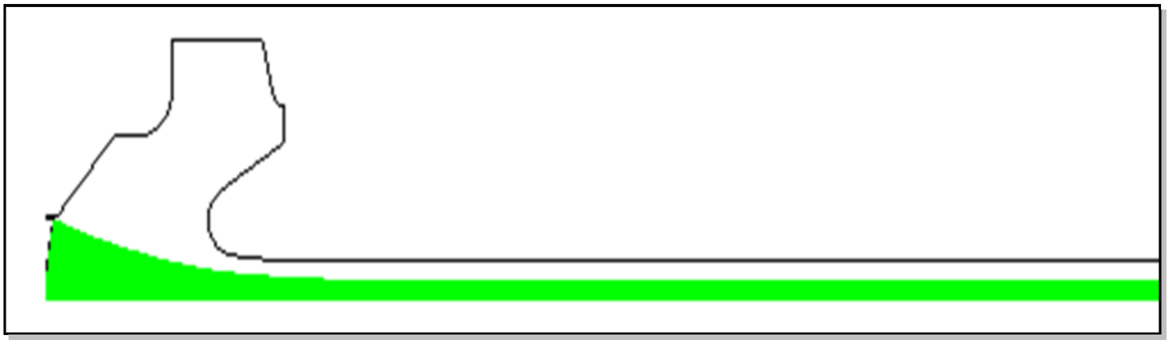


Fig. 15. Result of introducing magnetic field onto the result from step 2 (Fig. 13) and modifying the magnitude of the field and shifting the field along the axis to achieve minimum beam ripple with a given average beam radius.

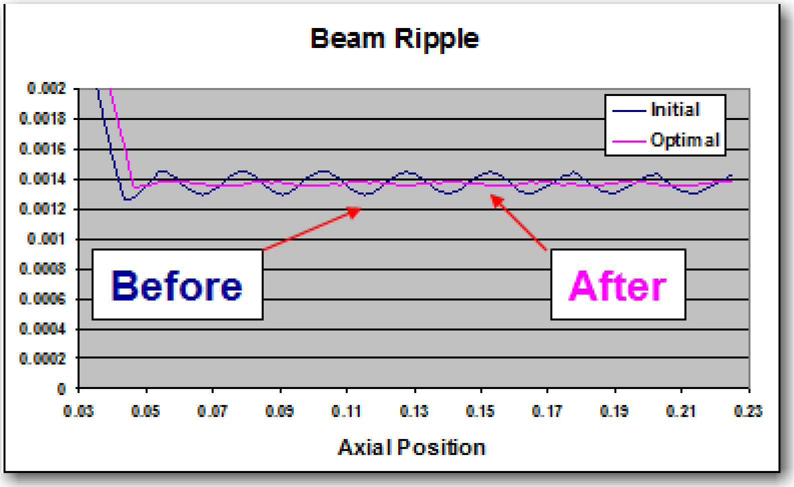


Fig. 16. A comparison of beam ripple for the initial and optimal design points.

Collector Optimization

Two collectors were investigated during the base project. The first problem was a 2D axisymmetric two-stage collector shown in Fig. 17.

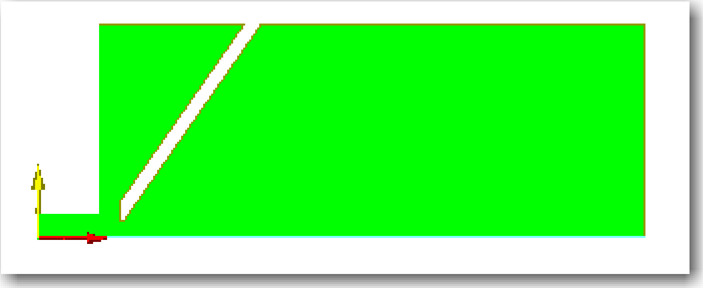
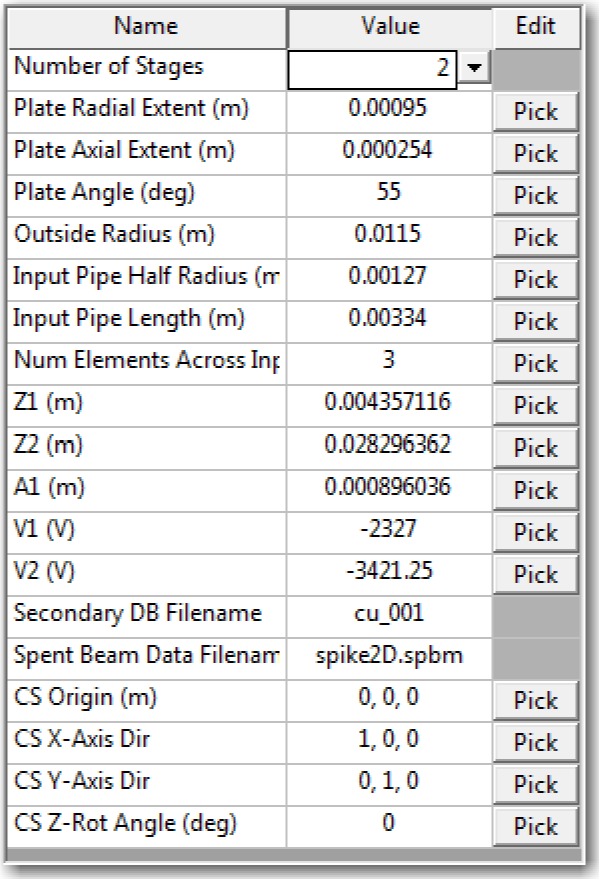


Fig. 17. Simple two stage collector that was optimized.

To facilitate this problem, a scripted solid capable of producing a simple collector geometry complete with boundary conditions was created. This scripted solid allows the user to control the number of stages (1-5), axial position of the stages, aperture sizes, and stage voltages (Fig. 18).

The goal of this optimization was to improve on the efficiency of the default geometry (current collector efficiency 76.39%) by first varying just the voltage of the 1st stage and then by varying the voltages of both stages. Both of these optimizations were performed using DE, MDS, and Powell to better understand how the various algorithms performed on collectors. Tables 9 and 10 illustrate the results. These optimizations were run until stagnation was detected. V1 was allowed to vary from -4000 V to -1000 V and V2 was allowed to vary from -6000 V to -2000 V.



Name	Value	Edit
Number of Stages	2	
Plate Radial Extent (m)	0.00095	Pick
Plate Axial Extent (m)	0.000254	Pick
Plate Angle (deg)	55	Pick
Outside Radius (m)	0.0115	Pick
Input Pipe Half Radius (m	0.00127	Pick
Input Pipe Length (m)	0.00334	Pick
Num Elements Across Inp	3	Pick
Z1 (m)	0.004357116	Pick
Z2 (m)	0.028296362	Pick
A1 (m)	0.000896036	Pick
V1 (V)	-2327	Pick
V2 (V)	-3421.25	Pick
Secondary DB Filename	cu_001	
Spent Beam Data Filenam	spike2D.spbm	
CS Origin (m)	0, 0, 0	Pick
CS X-Axis Dir	1, 0, 0	Pick
CS Y-Axis Dir	0, 1, 0	Pick
CS Z-Rot Angle (deg)	0	Pick

Fig. 18. Specification window for variable stage collector scripted solid.

In both optimizations, DE resulted in a better design point than MDS or Powell, but at a much higher cost. It is likely that both MDS and Powell fell into local minima, while DE eventually found the global minimum.

Table 9. Results of optimizing the two stage collector by varying only the voltage of the first stage. The initial value of this voltage was -2327 V.

	DE	MDS	Powell
Efficiency Improvement	2.37%	1.97%	**1.97%**
Number of Analyses	114	22	**3**
Overall Time (m)	67	12	**1.5**
Optimal V1	**-3012.45**	**-2787.25**	**-2792.4**

Next, a four stage collector model provided by L3-ETI was investigated at to provide experience with a production level device. The profile of this proprietary axisymmetric collector is quite complex (a facsimile is shown in Fig. 19). L3-ETI provided the geometry for the structure which was already manually optimized and had been tested experimentally on TWTs. STAAR created a scripted solid that allows for the variation of the axial positions, aperture radii, and voltages of several of the stages. In total, there were 8 (6 geometric and 2 voltage) parameters involved in the optimization process. The goal was to improve the collection efficiency. The application of these tubes to space qualified hardware means that even a small improvement of 1% in efficiency would be considered significant.

Table 10. Results of optimizing the two stage collector by varying voltages of both stages. The initial values of these voltages were -2327 V and -3421.25 V respectively.

	DE	MDS	Powell
Efficiency Improvement	6.78%	6.48%	**6.61%**
Number of Analyses	557	56	**61**
Overall Time (min)	325	33	**35**
Optimal V1	-3078.09	-3034.59	**-3046.52**
Optimal V2	**-3759.17**	**-3771.26**	**-3769.12**

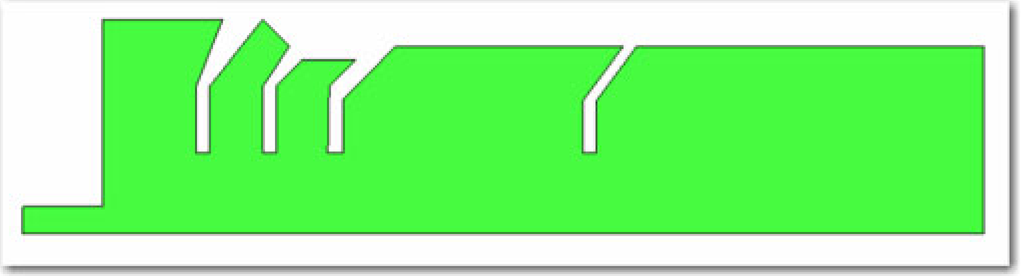


Fig. 19 . Facsimile of actual geometry.

A reasonable mesh (135k elements) was used resulting in each analysis (including only primaries and an axial magnetic field) taking approximately 4 minutes. Finer meshes were then run on the optimal design points to confirm the results. The results listed in Table 11 show that Powell, after 128 MICHELLE solutions, resulted in the best improvement in efficiency, 1.76%. This is a dramatic improvement over what was achieved manually.

Table 11. Results of optimizing L3-ETI four stage collector containing 6 geometric and 2 voltage parameters.

	DE	MDS	Powell
Efficiency Improvement	1.45%	1.50%	**1.76%**
Number of Analyses	298	128	**178**
Overall Time (hrs)	**19.9**	**8.5**	**11.8**

The initial and optimal design points were subsequently run using very fine meshes (1.6 million elements) to confirm the results of the optimization using a courser mesh. Table

12 illustrates minimal differences in the results of the 2 meshes. Note average runtimes for the courser mesh of 4 minutes and 50 minutes for the finer mesh.

Table 12. Effect on efficiency of various design points in moving from 135,000 elements to 1.6 million elements.

	Initial	DE Best	MDS Best	Powell Best
Absolute Change in Efficiency	0.02%	-0.05%	-0.20%	**-0.21%**
Corrected Efficiency Improvement	**-**	**1.47%**	**1.30%**	**1.55%**

The Powell solution (1.55% improvement) optimized design point has been provided to L3-ETI for further studies. It is anticipated that this design will be incorporated into their next production model.

Documentation

Substantial documentation was generated during this project covering all of the MICHELLE setup panels, the optimization process including descriptions of the methods, information on writing metrics and constraints, and several start-to-finish gun and collector examples. These additions will assist new and existing Analyst users in becoming familiar with MICHELLE and optimizations in general.