A PARALLEL MULTI-OBJECTIVE DIFFERENTIAL EVOLUTION ALGORITHM FOR PHOTOINJECTOR BEAM DYNAMICS OPTIMIZATION*

J. Qiang†, Y. Chao, C. E. Mitchell, S. Paret and R. D. Ryne, LBNL, Berkeley, CA 94720, USA

Abstract

In photoinjector design, there is growing interest in using multi-objective beam dynamics optimization to minimize the final transverse emittances and to maximize the final peak current of the beam. Most previous studies in this area were based on genetic algorithms. Recent progress in optimization suggests that the differential evolution algorithm could perform better in comparison to the genetic algorithm. In this paper, we propose a new parallel multi-objective optimizer based on the differential evolution algorithm for photoinjector beam dynamics optimization. We will discuss the numerical algorithm and some benchmark examples. This algorithm has the potential to significantly reduce the computation time required to reach the optimal Pareto solution.

INTRODUCTION

The photoinjector is a key component in the accelerator beam delivery system of next generation light sources by generating a high brightness electron beam into the accelerator. The goal of the photoinjector beam dynamics design is to achieve a high peak current while maintaining low transverse emittances at the same time. This requires optimizing a number of physical control parameters such as accelerating RF cavity amplitudes and phases, focusing solenoid strength and locations, and initial distribution of the electron beam. In previous studies, multi-objective optimization based on genetic algorithms has been used in the photoinjector beam dynamics optimization [1, 2, 3]. In this paper, we propose a new algorithm based on the differential evolution method for multi-objective beam dynamics optimization.

The differential evolution algorithm is a relatively new method in evolutionary algorithms [4]. It is a simple but powerful population-based, stochastic, direct-search algorithm with self-adaptive step size to generate next-generation offspring for global optimization. In a number of comparison studies, it has been shown to be more efficient than simulated annealing method, controlled transom search, evolutionary programming, and genetic algorithms [4, 5, 6]. It has been successfully used in a variety of applications and demonstrated its effectiveness [7, 8, 9]. Most of those algorithms are based on a fixed population size during the evolution. In this paper, we propose a parallel multi-objective differential evolution algorithm based on varying population size with external storage during the evolution.

DIFFERENTIAL EVOLUTION METHOD

In the differential evolution algorithm, a population with size NP in control parameter space is randomly generated at the beginning. This population is taken as the first generation of the control parameters. A new generation of the control parameter population is generated as follows: For each parameter vector $x_{i,G}, i = 0, 1, 2, \cdots, NP-1$ in a population of size NP at generation G, a perturbed vector v_i is generated according to

$$v_i = x_{i,G} + F_{CR}\,(x_{b,G} - x_{i,G}) + F_{xc}\,(x_{r1,G} - x_{r2,G}) \quad (1)$$

where the integers $r1$ and $r2$ are chosen randomly from the interval $[1, NP]$ and are different from the running index i, F_{xc} is a real scaling factor that controls the amplification of the differential variation $(x_{r1,G} - x_{r2,G})$, $x_{b,G}$ is the best parameter solution in generation G, and where F_{CR} is a combination weight factor between the original individual parent and the best parent. In most typical simulations, F_{xc} is set to 0.8, and F_{CR} can be chosen from a uniform random number between 0 and 1 or a fixed input number. In order to increase the diversity of the parameter vectors, crossover between the parameter vector $x_{i,G}$ and the perturbed vector v_i is introduced with an externally supplied crossover probability Cr to generate a new trial vector $U_{i,G+1}, i = 0, 1, 2, \cdots, NP-1$. For a D dimensional control parameter space, the new trial parameter vector $U_{i,G+1}, i = 0, 1, 2, \cdots, NP-1$ is generated using the following rule:

$$U_{i,G+1} = (u_{i1,G+1}, u_{i2,G+1}, \cdots, u_{iD,G+1}) \quad (2)$$

$$u_{ij,G+1} = \begin{cases} v_{ij}, & \text{if } rand_j \le CR \text{ or } j = mbr_i \\ x_{ij}, & \text{otherwise} \end{cases} \quad (3)$$

where $rand_j$ is a randomly chosen real number in the interval $[0, 1]$, and the index mbr_i is a randomly chosen integer in the range $[1, D]$ to ensure that the new trial vector contains at least one parameter from the perturbed vector. Next, the new trial solution $U_{i,G+1}$ is checked against the original parent $x_{i,G}$. If the new trial solution produces a better objective function value, it will be put into the next generation $(G+1)$ population. Otherwise, the original parent is kept in the next generation population. The above procedure is repeated for all NP parents to generate a new generation of population. This completes one iteration. Many iterations or generations are used to attain the final global optimal solution.

* Work supported by the U.S. Department of Energy under Contract No. DE-AC02-05CH11231.

† jqiang@lbl.gov

ISBN 978-3-95450-122-9

A PARALLEL DIFFERENTIAL EVOLUTION ALGORITHM FOR MULTI-OBJECTIVE OPTIMIZATION

The problem of multi-objective function optimization can be stated in the general mathematical form as:

$$min \begin{cases} f_1(x) \\ \cdots \\ f_n(x) \end{cases} \text{subject to } g_i(x) \leq 0, h_i(x) = 0 \quad (4)$$

Here, $f_1, \cdots, f_n$ are n objective functions to be optimized, x is a vector of control parameters, and g_i and h_i are constraints to the optimization. The goal of multi-objective optimization is to find the Pareto front in the objective function solution space. The Pareto optimal front is a collection of all non-dominated solutions in the whole feasible solution space. Any other solution in the feasible solution space will be dominated by those solutions on the Pareto optimal front. In the multi-objective optimization, a solution A is said to dominate a solution B if all components of A are at least as good as those of B (with at least one component strictly better). The solution A is non-dominated if it is not dominated by any solution within the group. An example of the Pareto front is shown as the green line within the feasible solution space in Fig. 1 with two objective functions.

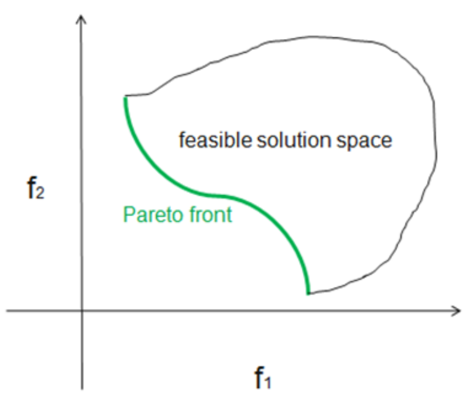

Figure 1: Feasible solution space and the Pareto optimal front in a two-objective function optimization.

The multi-objective differential evolution algorithm with varying population size in each generation and external storage can be summarized in the following steps:

- Step 0: Define the minimum parent size, $NPmin$ and the maximum size, $NPmax$ of the parent population. Define the maximum size of the external storage, $NPext$.
- Step 1: An initial population of $NPini$ parameter vectors is chosen randomly to cover the entire solution space.
- Step 2: Generate the offspring population using the above differential evolutionary algorithm.
- Step 3: Check the new population against boundary conditions and constraints.
- Step 4: Combine the new population with the existing parent population from external storage. Non-dominated solutions ($Ndom$) are found from this group of solutions and $min(Ndom, Next)$ of solutions are put back into external storage. Pruning is used if $Ndom > Next$. NP parent solutions are selected from this group of solutions for next generation production. If $NPmin \leq Ndom \leq NPmax$, $NP = Ndom$. Otherwise, $NP = NPmin$ if $Ndom < NPmin$ and $NP = NPmax$ if $Ndom > NPmax$. The elitism is emphasized through keeping the non-dominated solutions while the diversity is maintained by penalizing the over-crowded solutions through pruning.
- Step 5: If the stopping condition is met, stop. Otherwise, return to Step 2.

The above population based differential evolutionary optimization algorithm naturally leads to a multi-processor parallel implementation. Our method contains two levels of parallelization. First, the whole population is distributed among a number of groups of parallel processors. Each group of processors contains a subset of the whole population. Different sets of the sub-population can be tracked simultaneously. Second, each objective function evaluation corresponds to an accelerator simulation, for which parallel codes are available. Here, those objective function values such as transverse emittances and bunch length will be extracted from the results of parallel PIC beam dynamics simulations with a given accelerator lattice and beam parameters. A good scalability of the parallel differential evolution algorithm has been demonstrated on a Cray XT5 computer using $100,000$ processors in our previous study [10].

BENCHMARK EXAMPLES

As a test, we benchmarked the numerical solution using the above algorithm with two problems in reference [11]. The objective functions and the analytical solutions are given in Table 1. The numerical solutions together with the analytical solutions are given in Figs. 2 and 3. It is seen that the numerical solutions and the analytical solutions are in excellent agreement.

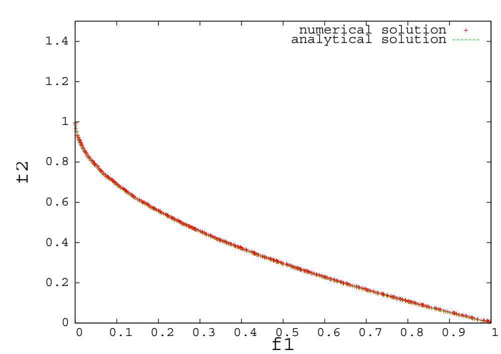

Figure 2: Pareto optimal front from the numerical solutions and the analytical solutions for problem ZDT4.

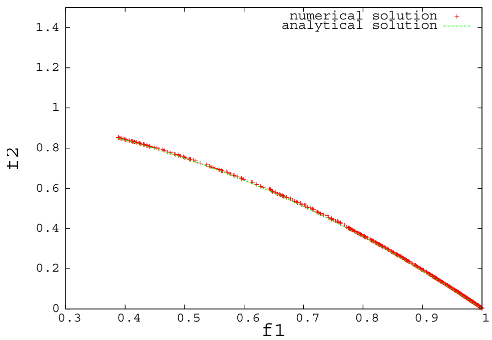

Figure 3: Pareto optimal front from the numerical solutions and the analytical solutions for problem ZDT6.

APPLICATION TO A PHOTOINJECTOR BEAM DYNAMICS OPTIMIZATION

As an application example, we used the above algorithm together with a particle-in-cell code [12] in a photoinjector beam dynamics optimization. A schematic plot of the photoinjector is shown in Fig. 4. It consists of a 187 MHz

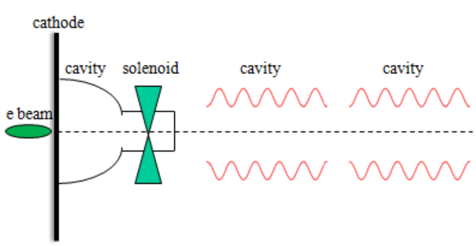

Figure 4: A schematic plot of a photoinjector for multi-objective optimization application.

RF gun, a solenoid, and two 650 MHz boosting cavities. The objective functions to be optimized are the transverse rms emittances and the longitudinal rms bunch length that is directly related to the peak current of the beam. There are 10 control parameters that are used in the optimization. Those are the initial electron beam transverse size and bunch length, RF gun phase, strength and location of the solenoid field, starting location of the boosting RF cavity, and amplitudes and phases of the two RF cavities. The maximum amplitude of the RF field inside the gun is set as about 19.5 MV/m. The charge of the electron beam is 300 pC. Some optimal solutions of the rm bunch length and the transverse emittance are shown in Fig. 5. It is seen that a transverse emittance emittance below 1 mm-mrad can be achieved with a reasonably short bunch length.

ACKNOWLEDGEMENT

This research used computer resources at the National Energy Research Scientific Computing Center and at the National Center for Computational Sciences.

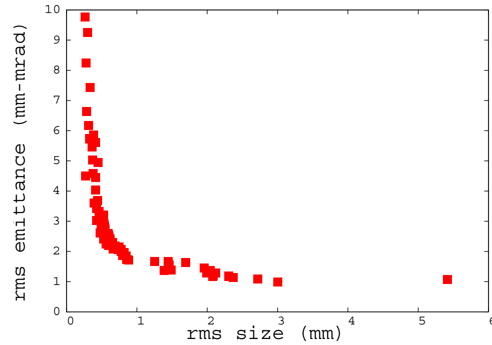

Figure 5: Optimal rms bunch length and transverse emittance solutions of the photoinjector beam dynamics optimization.

Table 1: Benchmark Problems [11]

problem	variable bounds	objective functions
ZDT4	$x_1 \in [0,1]$	$f_1(x) = x_1$
	$x_i \in [-5,5]$	$f_2(x) = g(x)[1 - \sqrt{x_1/g(x)}]$
	$i = 2, \cdots, n$	$g(x) = 1 + 10(n-1) + \sum_{i=2}^{n}[x_i^2 - 10cos(4\pi x_i)]$
ZDT6	$[0,1]$	$f_1(x) = 1 - \exp(-4x_1)sin^6(6\pi x_1)$
		$f_2(x) = g(x)[1 - (f_1(x)/g(x))^2]$
		$g(x) = 1 + 9[(\sum_{i=2}^{n} x_i)/(n-1)]^{0.25}$

REFERENCES

[1] I.V. Bazarov and C.K. Sinclair, Phys. Rev. ST Accel. Beams 8, 034202 (2005).

[2] C. E. Papdopoulos et al.,in Proceedings of FEL2010, p. 479.

[3] A. Hofler et al., Phys. Rev. ST Accel. Beams 16, 010101 (2013).

[4] R. Storn and K. Price, Journal of Global Optimization 1a1:341-359, (1997).

[5] K. Price, R. Storn, and J. Lampinen, Differential Evolution - A Practical Approach to Global Optimization, Springer, Berlin, 2005.

[6] M. M. Ali and A. Torn, Computers and Operations Research, Elsevier, no. 31, p. 1703, 2004.

[7] F. Xue, A. C. Sanderson and R. J. Graves. in Proceedings of CEC'2003, p. 862.

[8] B. V. Babu, P. G. Chakole, J. H. Syed Mubeen, Chemical Engineering Science, vol. 60, p. 4822 (2005).

[9] X. Wang, M. Hao, Y. Cheng, R. Lei, Journal of Universal Computer Science, vol. 15, p. 7222, (2009).

[10] J. Qiang and R. D. Ryne, in Proc. PAC2011, p. 1770.

[11] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, IEEE Trans. Evol. Comp, Vol. 6, p. 182, (2002).

[12] J. Qiang et al., Phys. Rev. ST Accel. Beams 9, 044204 (2006).

ISBN 978-3-95450-122-9