Energy Syst
DOI 10.1007/s12667-009-0008-3

ORIGINAL PAPER

One-stage R&D portfolio optimization with an application to solid oxide fuel cells

Lauren Hannah · Warren Powell · Jeffrey Stewart

Received: 26 May 2009 / Accepted: 26 December 2009

Abstract This paper provides an overview of the one-stage R&D portfolio optimization problem. It provides a novel problem model that can be solved with stochastic combinatorial optimization methods. Current solution methods are reviewed and a new method that scales to large problems, Stochastic Gradient Portfolio Optimization (SGPO), is proposed. Although SGPO is a heuristic method, we prove global convergence in certain conditions. SGPO is numerically compared to current optimization methods on a test case involving Solid Oxide Fuel Cells.

Keywords R&D portfolio · Stochastic combinatorial optimization · Solid oxide fuel cell

1 Introduction

It has become increasingly apparent that the current energy production and conversion technologies will be unsustainable in the future, but there are no clear technological successors. Research and development (R&D) is needed to reshape the energy infrastructure, but it is unclear how resources should best be spent. This paper considers a group of projects related to Solid Oxide Fuel Cells (SOFCs). Due to a budget constraint, only a subset of the projects may be funded, called a portfolio. Once they are funded, they produce a change in the technology parameters that are related to the

L. Hannah (✉) · W. Powell
Department of Operations Research and Financial Engineering, Princeton University, Sherrerd Hall, Princeton, NJ 08544, USA
e-mail: lhannah@princeton.edu

J. Stewart
Regional Energy System Technology Analysis Center, Lawrence Livermore National Laboratory, 7000 East Avenue, Livermore, CA 94551, USA

Published online: 21 January 2010

selected projects, such as the production cost or lifetime for a specific SOFC component. The collection of updated parameters are then used in a function that measures the overall value of the collection of technologies, most often as a cost of the best product that may be produced by those technologies. When a funding decision is made only once, the problem is referred to as a *one-stage R&D problem*. Often, an R&D problem will require funding decisions to be made in a sequential decision process (a *multi-stage R&D problem*), but that is beyond the scope of this paper.

This paper considers R&D portfolio optimization in an application to Solid Oxide Fuel Cells. Although creating an operable SOFC is significantly less challenging than creating a zero-greenhouse-gas-emission energy system, the cost function and project interactions related to SOFCs still demonstrate considerable complexity. A fuel cell is comprised of six components, with projects that produce attribute changes for individual components. The component-based project structure leads to highly non-linear project interactions, often in the form of a minimum function when we have to choose the lowest cost component. Project outcomes may have correlations between attributes within the project and sometimes when there are competing projects. A novel modeling method is proposed to capture the complexity derived from these issues. Computational difficulties come from the potentially huge number of portfolios, observations that provide only a noisy measurement of the portfolio's true value, and the convoluted correlations between portfolios. To manage these obstacles, this paper proposes a new algorithm, Stochastic Gradient Portfolio Optimization (SGPO), designed to search a large number of possible portfolios and offer an acceptable selection in finite time. Although this is a heuristic algorithm, we prove convergence to a global optimum under limited conditions.

Problems of this nature are addressed in both the R&D portfolio literature and combinatorial stochastic optimization literature. The R& D literature tries to solve this problem in various ways, often by limiting the form that the cost function may take. The combinatorial stochastic optimization literature does not deal specifically with the one-stage R&D portfolio problem, but it does address similar problems. Previous literature is reviewed in Sect. 2. Section 3 defines the problem with special reference to Solid Oxide Fuel Cells (SOFCs). A search heuristic based upon linearized marginal project values is presented in Sect. 5. Competing algorithms are discussed in Sect. 6. Numerical results are given in Sect. 7 and conclusions in Sect. 8.

2 Literature review

Work relevant to this problem comes in two main categories: R&D literature and stochastic combinatorial optimization literature. The R& D literature catalogues the previous approaches that have been taken solving the one-stage R&D portfolio problem, usually by redefining it into a problem easily solved by traditional methods. Stochastic combinatorial optimization is a subfield of global optimization where the cost function is noisy and the constraint set is combinatorial, usually integer. This description fits the one-stage R&D portfolio problem well, so solution methods in this field will also be surveyed.

2.1 R&D literature

Most papers written about the R&D portfolio problem involve one-stage portfolio selection. Although this problem is simpler than the multi-stage problem, a few factors still present difficulties: the decision space may be quite large, projects may have multiple attributes, outcomes are uncertain and can have non-trivial distributions, and projects may interact in complex ways.

Mathematical programming was one of the first approaches to the one-stage R&D portfolio problem and is still one of the most natural. Mathematical programming seeks to maximize a function, such as expected portfolio rewards, with respect to a decision, the portfolio selection, subject to constraints on the portfolio. The first applications were linear programming models of [25] and [6]. Linear programming is easy to implement and solve for large problems, but it assumes no project interactions, continuous decision variables, linear constraints and deterministic project contributions. References [48] and [49] give modern examples. Various authors have adapted mathematical programming to cope with particular attributes of the R&D portfolio problem. Reference [64] introduces a formulation with integer and other non-linear constraints and a non-linear cost function. References [57] and [10] further expand the types of cost functions used. References [38] and [61] use goal programming for projects with multiple attributes. More recent theoretical papers use a combination of these techniques [40, 54, 55], while industrial applications use mixed integer linear programs [8, 42]. Mathematical programming is well-studied and often quickly solvable for large problems, yet it is practical for only for certain cost function and constraint forms and it produces a deterministic solution.

Uncertainties have been modeled within a mathematical programming framework by stochastic programming and fuzzy programming. Stochastic programming modifies mathematical programming to explicitly incorporate probabilistic outcome scenarios. Surprisingly, it has been used only rarely in an R&D setting, see [14, 17, 43] and [36] for early examples. More recently, [23] use stochastic programming to model a multi-staged mixed portfolio of R&D projects and securities without any project interactions. It is also used much more extensively in pharmaceutical pipeline management than in traditional R&D portfolio management, see [19, 51]. Fuzzy programming is used to implicitly include uncertainty, by giving constraints and/or objective functions a set membership values between 0 and 1 instead of a traditional metrics. See [18, 63].

There has been a recent trend toward producing a set of Pareto-efficient portfolios of projects with multiple attributes, rather than a single selection. Methods for computing the efficient set vary, from branch and bound techniques [50] to ant colony optimization [20]. References [22, 59] present methods for deterministic portfolio values while [44] uses genetic algorithm search techniques to produce a stochastically non-dominated set of portfolios with uncertain values. Efficient portfolios are appealing because they allow the user to weight preferences after the portfolios are presented, but in practice the size of the efficient set can be unworkably large.

There have been isolated papers that attempt to cope with some of the difficulties of the R&D portfolio problem in ways other than those presented, but they often have restrictive assumptions. Reference [7] has a cost function based on the sums of maxima of competing projects, but computes the overall cost function expectation via the joint distributions of all non-separable project combinations. As a whole, the R&D literature has avoided describing the one-stage problem in full generality. The next subsection will describe it with full generality and present the current solution techniques for that problem.

2.2 Stochastic combinatorial optimization literature

A class of stochastic combinatorial optimization (SCO) problems, often called discrete stochastic optimization or subset selection problems, has the form

$$\begin{aligned} &\min \mathbb{E}C(x, W) \\ &\text{subject to } x \in \mathcal{X}, \end{aligned} \tag{1}$$

where W is a random variable, x is the decision variable, and $\mathcal{X}$ is a finite set. Let the decision x be such that $x_i = 1$ when project i is funded and 0 when it is not, $\mathcal{X} \subseteq \{0, 1\}^m$ be the set of all feasible portfolios out of m possible projects, and W be the technology parameter changes after funding, we describe the one-stage R&D portfolio problem in its most general form. The cost function C is assumed only to be defined over $\mathcal{X}$. Any method that solves (1) will solve the one-stage R&D portfolio problem.

Stochastic combinatorial optimization is a subset of global optimization. SCO methodology has received only moderate attention in the literature and no solution method is dominant for all problem sizes. Methods for low to medium-dimensional are more highly studied than those for higher dimensions. Most R&D problems are high dimensional; funding 15 projects out of a possible 30 results in more than 122 million potential portfolios and the number grows factorially with the total number of projects.

The smallest stochastic combinatorial optimization problems are treated as multiple hypothesis comparison problems. Methods were first presented by [62] and [41], and [35] gives a modern overview. Unlike most other SCO methods, multiple hypothesis testing has been used on the R&D portfolio problem (see [60]). Multiple hypothesis testing has the benefit of being well-studied, but it still requires all elements of $\mathcal{X}$ to be enumerated and sampled at least once. This is feasible when $|\mathcal{X}|$ is less than a few hundred.

In larger problems not every element of $\mathcal{X}$ can be sampled. Sampling decisions have been approached in a number of ways. Partitioning methods divide $\mathcal{X}$ into subsets and then generate bounds on those subsets, sequentially eliminating and reducing the size of each until a solution is reached. More commonly used, however, are search algorithms that rely at least partially on a random search of $\mathcal{X}$. These include simulated annealing (SA), ant colony optimization (ACO), and evolutionary computation (EC). The following is a brief overview of contemporary SCO methods; see [9] for a more complete review.

Partitioning methods rely on dividing $\mathcal{X}$ and sampling the most promising portions. The branch and bound technique is applied to stochastic problems by [46], who provide a proof of convergence. The Nested Partitions method is proposed by [56]. It samples partitions based on a Markov chain structure and chooses a solution based on the number of times each singleton partition was visited. These methods tend to have high computation times for large problems.

There have been a number of SCO algorithms that involve random searches. Many use pairwise decision comparison to produce a sequence of increasingly good solutions, such as [2, 4, 5, 65], and [3]. Convergence is shown in some cases by increasing the sampling sizes or keeping a record of estimates for all decisions sampled. Ordinal optimization expands random searches to high dimensions by softening the procedural goal from finding the optimal solution to finding an acceptable solution with high probability (see [15, 16, 33]).

Simulated annealing is one of the most widely studied SCO random search algorithms. Based on the work of [45], [39] and [11] independently suggested the method for deterministic global optimization. References [1, 24, 26, 31] and [47] extend simulated annealing to the stochastic problem and show convergence for certain cases with discrete decision sets. Converge can be quite slow, especially in high dimensions.

Evolutionary Computation and Ant Colony Optimization algorithms vary from the above random search algorithms in that they search a population of decisions, rather than a single, competing alternative. EC relies on evolutionarily and genetically inspired search mechanisms, such as mutation and survival of the fittest. Reference [34] proposed Genetic Algorithms, the evolutionary computation subfield most commonly used in stochastic combinatorial optimization. Although convergence has been shown under certain conditions for deterministic functions on finite, combinatorial decision spaces [29, 53], it is not inherent [52]. Reference [37] gives a survey of EC applied to noisy environments. Recent work by [13] and [12] has produced provably convergent algorithms for solving Markov Decision Processes. Reference [32] extend this work to solving problems with the form of (1). Ant Colony Optimization, introduced by [21], uses random paths and time-dependent "pheromone trails" to find good solutions. Reference [30] gave a proof of convergence for SCO-type problems.

While many of the above mentioned algorithms are provably convergent, they are not made for large problems. All either rely on decision space enumeration or a random search of that decision space. Structure-based searches should reach a reasonable solution more quickly than random searches, especially for large problems. There have been some attempts to use structure-based searches, such as an extension of the Simultaneous Perturbation Stochastic Approximation of [58] to discrete decision spaces by [27]. It, however, cannot be used for all discrete decision spaces. We seek to provide a structure-based search for large R&D portfolio problems.

3 Problem formulation

This paper considers how resources can be allocated across a variety of projects necessary for a Solid Oxide Fuel Cell (SOFC). Fuel cells provide combustion-less conversion of hydrogen into electricity. H_2 contacts the anode and is split into positive hydrogen ions and electrons. Air contacts the cathode, which combines the electrons generated by the anode and the O_2 contained in the air to form negatively charged oxygen ions. The oxygen ions travel through the electrolyte, attach to the hydrogen

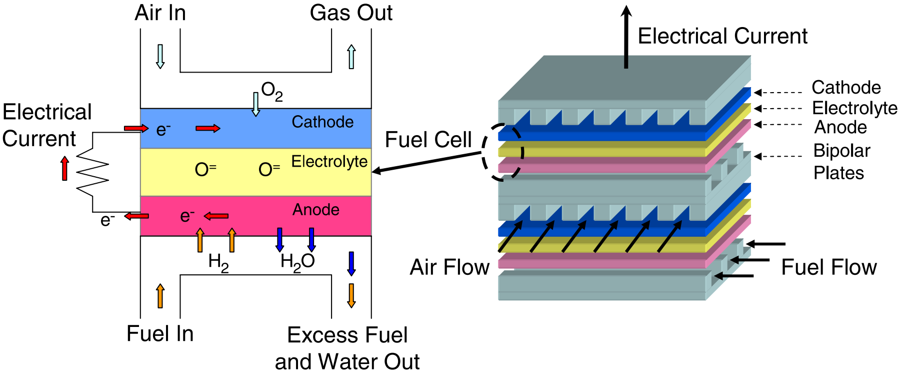

Fig. 1 A diagram of a Solid Oxide Fuel Cell (*left*) and a fuel cell stack (*right*)

Table 1 Components of a solid oxide fuel cell and associated technology parameters. Power density is given in W/cm^2

Component	Technology	Param.	Technology	Param.	Technology	Param.
Anode	Surface Area	ρ_A^{SA}	Power Density	ρ_A^{PD}	Prod. Cost	ρ_A^{COST}
Cathode	Surface Area	ρ_C^{SA}	Power Density	ρ_C^{PD}	Prod. Cost	ρ_C^{COST}
Electrolyte	Reaction Stability	ρ_E^{RS}	Degradation	ρ_E^{DEG}	Prod. Cost	ρ_E^{COST}
Bipolar Plates	Temp. Stability	ρ_B^{TS}	Conductivity	ρ_B^{CON}	Prod. Cost	ρ_B^{COST}
Seal	Temp. Stability	ρ_S^{TS}	Chemical Stability	ρ_S^{CS}	Prod. Cost	ρ_S^{COST}
Pressure Vessel	Design	ρ_P^{DES}	–	–	Prod. Cost	ρ_P^{COST}

ions and form water. The current is generated by the flow of electrons from the anode to the cathode. Fuel cells are characterized by their electrolyte; SOFCs have a solid oxide, or ceramic, electrolyte. Unlike other fuel cells, SOFCs can operate at high temperatures (up to 1,000 degrees C) and can utilize many gaseous fuels containing H_2, such as methane, propane, and gasified biomass, instead of pure H_2.

To create higher voltage, the individual fuel cells are placed in series with bipolar plates in between them. This configuration is known as a stack. Stacks are placed in a pressure vessel, which is then sealed to retain heat. See Fig. 1 for a diagram of a fuel cell.

An SOFC is comprised of six components: an anode, cathode, electrolyte, bipolar plate, seal and pressure vessel. As shown in Table 1, most components, with the exception of the pressure vessel, are characterized by three attributes. The values of the attributes are denoted by technology parameters, ρ. A project (in the form of a research proposal) may improve some or all of the components within a single component area. There may be multiple projects that might inimprove the same component, such as an anode created from metal A and an anode created from metal B. Projects for the same component are viewed as competing. If anode A is funded, it changes only the technology parameters associated with anode A. If a project is not funded the technology parameters retain their original values. There are m possible projects, out of which a subset are selected to form a portfolio. The selected projects are then funded, and a random change occurs in the technology parameters for those projects. After the change occurs, the results are measured by determining the cost of the best fuel cell that can be created. The combination of a single anode, cathode, electrolyte, bipolar plate, seal and pressure vessel that produces the lowest cost per unit time is chosen from all projects, funded or not. The model and cost function are described in more detail in the following section.

Each project i consists of a set of technology parameters,

$$\rho_{t,i} = \left(\rho^1_{t,i}, \rho^2_{t,i}, \ldots, \rho^{r_i}_{t,i}\right),$$

where t is a time index in $\{0, 1\}$ and $\{1, 2, \ldots, r_i\}$ are the technology numbers. In general, t denotes the time at which a random variable or function is known. The number r_i denotes the number of technology parameters, or attributes, for project i. For example, project i may represent a proposal to fund research in materials science to improve the coating on bipolar plates, which would influence the temperature stability, conductivity and production cost of the associated plate. Therefore, the technology parameters ρ^{TS}_i, ρ^{CON}_i, ρ^{COST}_i would be affected if the project were funded. For this application, alpha-numeric combinations such as $C2$ will be used to denote the project component and number due to the component-based nature of the problem, and a brief description, such as $COST$, will replace the generic technology index. The overall state of the technology parameters at time t is given by $\rho_t = (\rho_{t,1}, \rho_{t,2}, \ldots, \rho_{t,m})$. Table 1 lists all components and technology parameters relevant to those components.

Project technology parameters are indexed by the time, component and project number in the subscript with the technology in the superscript. For example, the parameter for the production cost of a cathode under project 2 at time 0 would be given by $\rho^{COST}_{0,C2}$. The entire state of technology for the cathode under project 2 at time 0 would be $\rho_{0,C2} = (\rho^{SA}_{0,C2}, \rho^{PD}_{0,C2}, \rho^{COST}_{0,C2})$. A hierarchical chart of the fuel cell/component/project/technology parameter structure is given in Fig. 2.

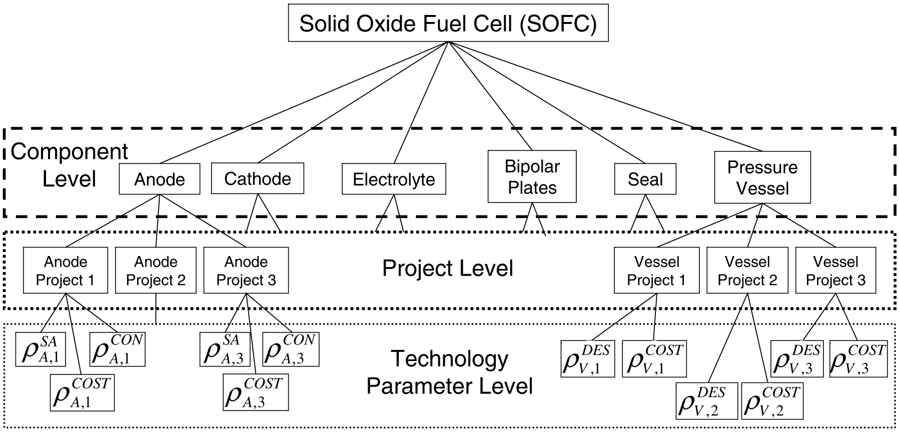

Fig. 2 Funding is done at the project level while the funding results in changes at the technology parameter level. The component level exists to help with modeling the cost function

Funding decisions are described by the vector x. If project i is funded, $x_i = 1$, otherwise, $x_i = 0$. The decision variable is an m-dimensional vector of 0's and 1's, representing the decision to fund or not to fund each of the m projects $x = (x_1, x_2, \ldots, x_m)$. Furthermore, each project i costs c_i to fund. Projects are then selected subject to a budget b, producing the constraint

$$\sum_{i=1}^{m} c_i x_i \leq b.$$

Once a project is funded, the technology parameters it affects may change. $\hat{\rho}_i^j$ is the change in the j^{th} technology area of the i^{th} project from the 0^{th} to the 1^{st} time period as a result of funding the research. The technology changes for project i are $\hat{\rho}_i = (\hat{\rho}_i^1, \hat{\rho}_i^2, \ldots, \hat{\rho}_i^{r_i})$. The technology change is a random variable dependent on ρ_0 and x. In the case of SOFCs, all exogenous information is assumed to be uniformly distributed. For example, the change in the production cost of a cathode of type 2 has the distribution

$$\hat{\rho}_{C2}^{COST} \sim x_i \times \mathrm{U}\big[0, \tilde{\rho}_{C2}^{COST}\big],$$

where $\tilde{\rho}_{C2}^{COST}$ is a fixed parameter denoting the upper bound for changes and $\mathrm{U}[a, b]$ is the uniform distribution on $[a, b]$. Although the above equation simply multiplies the decision with the change, the exogenous information is simply a function of the decision and some base parameters. It can include externally generated technology changes or correlations between technology parameters and projects. The value of the technology parameters at time 1 is given by

$$\rho_{1,i} = \rho_{0,i} + \hat{\rho}_i.$$

Once the new technology parameters have been obtained, a cost metric is used to determine their value. In the case of SOFCs, the goal is to produce the cheapest fuel cell per unit of average lifetime that still meets certain requirements on pressure vessel design and temperature stability. The cost is given by selecting the best anode, cathode, electrolyte, bipolar plates, seal and pressure vessel, computing the production cost, dividing by the lifetime of the whole fuel cell, and then adding a penalty for unmet specifications. The cost metric is given explicitly in the next section. The cost for a given portfolio x and outcome $\hat{\rho}$ is denoted as $C(\rho_1(\rho_0, x, \hat{\rho}))$.

The preceding elements are incorporated to formulate the one-stage R&D portfolio problem as

$$\begin{aligned} &\min_x \mathbb{E}\, C(\rho_1(\rho_0, x, \hat{\rho})) \\ &\text{subject to} \sum_{i=1}^{m} c_i x_i \leq b, \\ &x_i \in \{0, 1\}. \end{aligned} \tag{2}$$

While this may be very simple to write, it is quite hard to solve. This complexity comes from three main areas. First, the decision set, the region of feasible portfolios, is combinatorial and may be huge. Second, the expectation can only be approximated by Monte Carlo methods. Third, the cost function may have many local minima. The combination of these issues makes the use of most local or global optimization methods inefficient.

4 Cost function

Like many engineering problems, SOFC R&D portfolio optimization has a non-trivial cost function. We are concerned with finding the cost of the cheapest fuel cell available given components and their associated technology parameters. This section derives a cost metric C for a fuel cell, based on the capital cost of the SOFC per kWh. This metric does not include fuel costs. It assumes that the SOFC is built around a stack of 1,000 fuel cells, each with a footprint of 1,000 cm^2.

This metric will produce an estimate four quantities: the power output of the fuel cell, the lifetime, the cost and penalty terms. A final cost will be given in terms of $\$/kWh$ with penalties added afterward. Penalties are given for failure to meet certain criteria, such as a suitably wide operating temperature range and an adequate pressure vessel design metric.

The cost metric at time t will be calculated for each set of projects that is comprised of an anode, a cathode, an electrolyte, a bipolar plate, a seal and a pressure vessel. Denote this set by its indices, $\{i_1, i_2, i_3, i_4, i_5, i_6\} = \mathcal{I}$, where i_1 is the i_1^{st} anode project, i_2 is the i_2^{nd} cathode project, etc. The cost for the set of projects $\mathcal{I}$ is calculated in the following steps.

First, the power of the entire stack is calculated. Since the power density is determined by the minimum power density of the anode and cathode,

$$C_{t,\mathcal{I}}^{PD} = \min\{\rho_{t,Ai_1}^{PD} \times \rho_{t,Ai_1}^{SA}, \rho_{t,Ci_2}^{PD} \times \rho_{t,Ci_2}^{SA}\},$$
$$C_{t,\mathcal{I}}^{kW} = 1{,}000\ \text{cm}^2 \times C_{t,\mathcal{I}}^{PD} \times \rho_{t,Bi_4}^{CON},$$

where ρ_{t,Bi_4}^{CON} is the proportion of power lost through imperfect conductivity. The capital cost for the fuel cell is

$$C_{t,\mathcal{I}}^{COST} = 1{,}000 \times \left(\rho_{t,Ai_1}^{COST} + \rho_{t,Ci_2}^{COST} + \rho_{t,Ei_3}^{COST} + \rho_{t,Bi_4}^{COST}\right) + \rho_{t,Si_5}^{COST} + \rho_{t,Pi_6}^{COST}.$$

The lifetime of the fuel cell is the minimum of the lifetimes of the components,

$$C_{t,\mathcal{I}}^{LIFE} = \min\{\rho_{t,Ei_3}^{RS}, \rho_{t,Ei_3}^{DEG}, \rho_{t,Si_5}^{CS}\}.$$

The unpenalized capital cost per kWh is calculated by

$$C_{t,\mathcal{I}}^{COST} = \frac{C_{t,\mathcal{I}}^{COST}}{C_{t,\mathcal{I}}^{kW} \times C_{t,\mathcal{I}}^{LIFE}}.$$

The upper temperature bound is the minimum of all of the temperatures

$$C_{t,\mathcal{I}}^{TEMP} = \min\{\rho_{t,Bi_4}^{TS}, \rho_{t,Si_5}^{TS}\}.$$

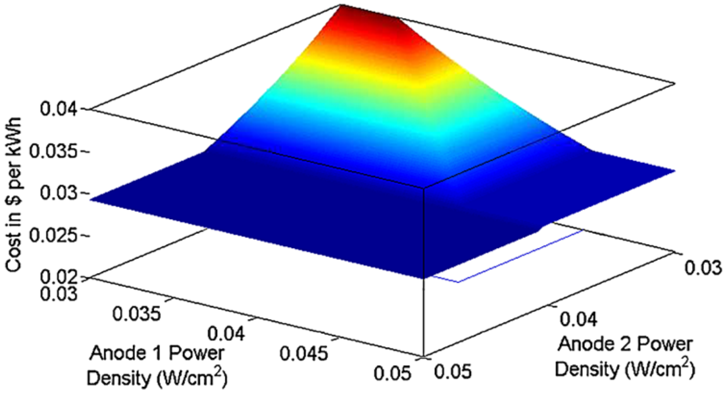

Fig. 3 The cost of a fuel cell with respect to varying power density in two competing anodes, with all other values held constant. Note that the cost function is not convex in these two variables

The penalties for not meeting design specifications and temperature range are calculated as follows,

$$C_{t,\mathcal{I}}^{PEN} = a^{DES}\left[\tilde{\rho}^{DES} - \rho_{t,Pi_6}^{DES}\right]^{+2} + a^{TEMP}\left[C_{t,\mathcal{I}}^{TEMP} - \tilde{\rho}^{TEMP}\right]^{+2},$$

where $\tilde{\rho}^{DES}$, $\tilde{\rho}^{TEMP}$ are threshold parameters and a^{DES}, a^{TEMP} are scaling parameters. Finally, the penalty is added to the unpenalized cost to get a cost for a specific component combination $\mathcal{I}$ at time t:

$$C_{t,\mathcal{I}} = C_{t,\mathcal{I}}^{COST} + C_{t,\mathcal{I}}^{PEN}.$$

The cost of a portfolio selection is defined as the lowest cost metric of a set of components at time 1 given portfolio selection x and the technology parameter realization $\rho_1(x,\omega)$. Therefore,

$$C(\rho_1(x,\omega)) = \min_{\mathcal{I}\in\mathcal{A}} C_{1,\mathcal{I}},$$

where $\mathcal{A}$ is the set of all feasible component combinations.

The function produced by this cost function lacks many desirable properties, such as convexity in the technology parameters. See Fig. 3 for a cut across the power densities of two competing anodes.

5 Stochastic gradient-based portfolio optimization

Most global optimization methods use some sort of randomized search to find portfolios (see [13, 31] and [3]). This may be necessary when the cost function has no structure, but it results in slow search progress when the decision space is large. We propose an algorithm that provides high quality solutions, with a high degree of reliability, while accommodating the full generality of the problem. We use a linear approximation of the objective function to produce a sequence of knapsack problems, which are easily solved to near optimal solutions by most modern math programming packages. The method scales to large values of m.

5.1 The SGPO algorithm

In (2), $\mathbb{E}C(\rho_1(\rho_0, x, \hat{\rho}_1))$ is impossible to compute exactly. Instead, it will be approximated by a function $V(x)$ that assigns marginal values, v_i, to each project. The approximated value of the portfolio is the sum of the marginal values of the projects in the portfolio,

$$V(x) = \sum_{i=1}^{m} x_i v_i.$$

The marginal values are estimated using a series of stochastic gradients. Special care needs to be taken due to the integrability constraints of the R&D portfolio problem, that is $x_i \in \{0, 1\}$ for $i = i, \dots, m$. If we allowed the constraints $0 \le x_i \le 1$ instead, we would be able to evaluate stochastic derivatives more and more locally, where

$$\hat{v}^{n+1} = \frac{C(x^n + \Delta^n) - C(x^n)}{\Delta^n}.$$

As we let $\Delta^n \to 0$, $\hat{v}^{n+1}$ would converge to a local stochastic gradient. Algorithms for such cases have already been presented in the literature, such as simultaneous perturbation stochastic approximation of [58].

Suppose that the values are currently estimated by $\bar{v}_1^n, \bar{v}_2^n, \dots, \bar{v}_m^n$ and that a feasible portfolio x^n has been chosen. Let ω^{n+1} be the $n + 1^{st}$ outcome and

$$v^{n+1} = C\big(\rho_1(\rho_0, x^n, \hat{\rho}_1(\omega^{n+1}))\big). \tag{3}$$

In order to determine the i^{th} stochastic gradient, we need to create a portfolio perturbed around the i^{th} project. Let $\hat{x}^{n,i}$ be a new portfolio found by solving

$$\begin{aligned} \hat{x}^{n,i} &= \arg\max_{y} \sum_{j=1}^{m} \bar{v}_j^n y_j \\ &\text{subject to} \sum_{j=1}^{m} c_j y_j \le b \\ &\qquad y_i = 1 - x_i^n \\ &\qquad y_j \in \{0, 1\}, \;\; j \ne i. \end{aligned} \tag{4}$$

This adds project i into the perturbed portfolio if it is not in the original and takes it out if it is. The perturbed technology change is found by

$$\hat{\rho}_j^{n+1,i,k} = \hat{x}_j^{n,i} \times U_j^k(\omega^{n+1}),$$

where $U_j^k \sim \mathrm{U}[0, \tilde{\rho}_j^k]$ is a random variable that describes the technology change of the k^{th} technology of the j^{th} project if that project were funded. The updated technology parameters for the perturbed portfolio are $\rho_1^{n+1,i} = \rho_0 + \hat{\rho}^{n+1,i}$. The value of the perturbed portfolio for the given outcome ω^{n+1} is

$$\hat{v}_i^{n+1} = \begin{cases} C(\rho_1^{n+1,i}) - v^{n+1} & \text{if } x_i^n = 1, \\ v^{n+1} - C(\rho_1^{n+1,i}) & \text{if } x_i^n = 0. \end{cases}$$

The n^{th} estimate for the i^{th} marginal value is obtained by smoothing the $n+1^{st}$ stochastic derivative realization into the preceding estimate using

$$\bar{v}_i^{n+1} = \alpha_n \hat{v}_i^{n+1} + (1-\alpha_n)\bar{v}_i^n, \tag{5}$$

where α_n is a stepsize in (0, 1]. The stepsize must meet some basic criteria,

$$\alpha_n > 0, \qquad (\alpha_n) \to 0, \qquad \sum_{n=0}^{\infty} \alpha_n = \infty. \tag{6}$$

We initialize the values of $\bar{v}$ by randomly choosing feasible portfolios for the first M iterations. If $n > M$, x^n is found by solving

$$\begin{aligned} x^n = \arg\max \sum_{i=1}^{m} x_i \bar{v}_i^n \\ \text{subject to} \sum_{i=1}^{m} c_i x_i \le b, \\ x_i \in \{0, 1\}. \end{aligned} \tag{7}$$

The first M iterations train $\bar{v}$ to a reasonable value.

The overall value function is updated by

$$\bar{V}^{n+1}(\rho_0) = \alpha_n \hat{v}^{n+1} + (1-\alpha_n)\bar{V}^n(\rho_0).$$

As in (5), α_n is a stepsize in (0, 1] and must satisfy the conditions in (6), but it is not necessarily the same as the one in (5). The complete algorithm is presented in Algorithm 1.

5.2 Convergence analysis for SGPO

In certain cases, SPGO can be guaranteed to converge to an optimal solution. The non-continuous nature of $\mathbb{E}(\hat{v}|\bar{v}) - \bar{v}$ in $\bar{v}$ makes traditional ODE convergence methods hard to use, but a few special cases have provable convergence.

We consider the case where there exists an ordering of marginal values that holds for all portfolio choices and a fixed budget b that must be satisfied. We use the subscripts $[1], [2], \ldots, [m]$ to denote the indices of the project with the largest marginal value ordered, [1], down to the project with the smallest marginal value, $[m]$.

Theorem 1 *Algorithm 1 converges to the optimal solution in probability if*

(a) *the project costs are equal, that is*

$$\sum_{i=1}^{m} x_i = b,$$

Algorithm 1: Stochastic Gradient Portfolio Optimization

```
Initialize $\bar{v}^0$
for $n = 0$ to $N$ do
    Choose a portfolio $x^n$:
    if $n \leq M$ then
        Pick random $x^n \in \{0, 1\}^m$
    else
        Find $x^n$ by solving (7)
    end if
    Get outcome $\omega^n$
    Set $\hat{v}^{n+1} = C(\rho_1(\omega^{n+1}, x^n))$
    for $j = 1$ to $m$ do
        Make a new portfolio $\hat{x}^{n,i}$ by solving (4)
        Find $\hat{v}_i^{n+1}$ by (3)
        $\bar{v}_i^{n+1} = \alpha_n \hat{v}_i^{n+1} + (1 - \alpha_n)\bar{v}_i^n$
    end for
end for
```

(b) *and there exists a set of indices* $[1], [2], \ldots, [m]$ *such that for any* $\bar{v}$,

$$\mathbb{E}(\hat{v}_{[1]}|\bar{v}) > \mathbb{E}(\hat{v}_{[2]}|\bar{v}) > \cdots > \mathbb{E}(\hat{v}_{[m]}|\bar{v}).$$

Proof To show convergence in probability, we need to show that for every $\epsilon > 0$ and $\bar{v}^0$ there exists N such that for every $n \geq N$, $\mathbb{P}\{\bar{v}^n_{[1]} > \bar{v}^n_{[2]} > \cdots > \bar{v}^n_{[m]}|\bar{v}^0\} > 1 - \epsilon$. First consider how $\bar{v}^n_{[k]}$ is estimated,

$$\bar{v}^n_{[k]} = \prod_{i=0}^{n-1}(1 - \alpha_i)\bar{v}^0_{[k]} + \sum_{i=1}^{n-1}\alpha_{i-1}\prod_{j=i}^{n-1}(1 - \alpha_j)(\hat{v}^i_{[k]}|\bar{v}^{i-1}) + \alpha_{n-1}(\hat{v}^n_{[k]}|\bar{v}^{n-1}),$$

where $\hat{v}^n_{[k]}|\bar{v}^{n-1}$ is the stochastic gradient at time n conditioned on the estimated marginal values (and hence portfolio) at time $n - 1$. $\bar{v}^n_{[i]}$ is the weighted sum of an initial term and n stochastic observations.

Fix $\epsilon > 0$. Using Boole's Inequality,

$$\mathbb{P}\big\{\bar{v}^n_{[1]} > \cdots > \bar{v}^n_{[m]}\big\} \geq 1 - \sum_{i=2}^{m}\mathbb{P}\big\{\bar{v}^n_{[i]} > \bar{v}^n_{[i-1]}\big\}.$$

We therefore want to show that we can choose n such that $\mathbb{P}\{\bar{v}^n_{[i]} > \bar{v}^n_{[i-1]}\} \leq \frac{\epsilon}{m}$ for each $i = 2, \ldots, m$. Let $\mu^n_i = \mathbb{E}(\bar{v}^n_{[i-1]} - \bar{v}^n_{[i]})$. Let

$$\mu_{min} = \min_{[i],[j],\bar{v}} |\mathbb{E}(\hat{v}_{[i]} - \hat{v}_{[j]}|\bar{v})|,$$

that is, the minimum distance between any two expected marginal values given any ordering. Furthermore, let

$$\sigma^2_{max} = \max_{[i],[j],\bar{v}} \operatorname{Var}(\hat{v}_{[i]} - \hat{v}_{[j]}|\bar{v}),$$

making σ^2_{max} the largest possible variance for the difference between any two marginal observations under any set of estimated values. Fix $\delta < \mu_{min}$. Choose n such that $\delta > \prod_{k=0}^{n-1}(1-\alpha_k)|\bar{v}^0_{[i]} - \bar{v}^0_{[j]}|$ for every pair $([i],[j])$. Then, we look at each inequality individually. By the one-sided Chebyshev Inequality,

$$\begin{aligned}\mathbb{P}\big\{\bar{v}^n_{[i]} - \bar{v}^n_{[i-1]} \geq 0\big\} &= \mathbb{P}\big\{\bar{v}^n_{[i]} - \bar{v}^n_{[i-1]} + \mu^n_i \geq \mu^n_i\big\} \\ &\leq \mathbb{P}\big\{\bar{v}^n_{[i]} - \bar{v}^n_{[i-1]} + \mu^n_i \geq \mu_{min} - \delta\big\} \\ &\leq \frac{\operatorname{Var}(\bar{v}^n_{[i]} - \bar{v}^n_{[i-1]})}{\operatorname{Var}(\bar{v}^n_{[i]} - \bar{v}^n_{[i-1]}) + (\mu_{min} - \delta)^2} \\ &\leq \frac{\lambda^n \sigma^2_{max}}{\lambda^n \sigma^2_{max} + (\mu_{min} - \delta)^2},\end{aligned} \tag{8}$$

where

$$\lambda^n = \begin{cases} (\alpha_{n-1})^2 & \text{if } n = 1, \\ (1-\alpha_{n-1})^2 \lambda^{n-1} & \text{if } n > 1. \end{cases}$$

Note that $\lambda^n \to 0$ as $n \to \infty$, so it is possible to choose n such that

$$\frac{\lambda^n \sigma^2_{max}}{\lambda^n \sigma^2_{max} + (\mu_{min} - \delta)^2} \leq \frac{\epsilon}{m}.$$

Since $\mathbb{P}\{\bar{v}^n_{[i]} - \bar{v}^n_{[i-1]} \geq 0\}$ is decreasing in n, for every $n \geq N$,

$$\mathbb{P}\{\bar{v}^n_{[1]} > \bar{v}^n_{[2]} > \cdots > \bar{v}^n_{[m]} | \bar{v}^0\} > 1 - \epsilon. \; \square$$

A number of problem formulations fulfill the conditions for Theorem 1, assuming that the expected cost of each portfolio has finite mean and variance. This includes the case where only one project is selected (with any finite mean and variance cost function) and the case where the cost function is separable (additive),

$$C(\rho_1) = \sum_{i=1}^{m} f_i(\rho_{1,i}).$$

6 Competing algorithms

The most common stochastic combinatorial optimization techniques for medium to large sized problems are simulated annealing (SA) and evolutionary algorithms.

Since evolutionary algorithms come in so many forms, we have chosen one that shows the most promise for our problem, Evolutionary Policy Iteration.

Simulated annealing can be made provably convergent via sampling routines. References [24, 26] and [31] all require that the noise in the estimates decrease each iteration. For most distributions of noise, this requires sampling the competing portfolios at least $\mathcal{O}(n^{\gamma})$, $\gamma > 2$, times each iteration. References [1] and [47] do not require an increasing number of samples each iteration, but they do require that averages for every portfolio visited be recorded. In practice, this is problematic because convergence requires that every portfolio be visited and have a corresponding estimate stored. The latter method is impractical for the R&D portfolio problem, so only the former will be used for numerical comparisons.

Evolutionary Policy Iteration (EPI) was proposed in [13] and [12]. It was suggested as a method to find optimal policies in Markov decision processes (MDP). The word "policy" refers to a set of state-action choices, but in our case a "policy" is simply a rule for choosing a portfolio. Since there is only one decision to be made and it depends on ρ_0, there is only one state, ρ_0. In essence, each iteration the algorithm has an "elite" portfolio choice from the previous iteration. A set of k portfolios is generated based on the elite portfolio, either as a relative of the elite portfolio with probability p, or as a mutation with probability $1 - p$. They are then compared, and the best one is chosen to be the new elite portfolio. This converges to the global minimum. The basic algorithm, modified from [13] to fit an MDP with a single starting state and a single time period, is given in Algorithm 2.

Algorithm 2: Evolutionary Policy Iteration (Chang et al. 2005)

```
Initialize $x^0$, specify $p$
**for** $n = 1$ to $N$ **do**
    Choose $\pi^n$, a population of portfolios:
    **for** $i = 1$ to $k$ **do**
        **if** $u \leq p$, where $u \sim \mathrm{U}[0, 1]$ **then**
            Choose $\pi_i^n$ as a random portfolio in $\mathcal{N}(x^{n-1})$, where $\mathcal{N}(x^{n-1})$ is a
            neighborhood around $x^{n-1}$
        **else**
            Choose $\pi_i^n$ as a random portfolio in $\mathcal{X}$
        **end if**
    **end for**
    Set $\pi_{k+1}^n = x^{n-1}$
    **for** $i = 1$ to $k + 1$ **do**
        Set $v_i^n = \mathbb{E}C(\rho_1(\pi_i^n))$
    **end for**
    Set $x^n = \pi_i^n$, $i = \arg\min_j v_j^n$
**end for**
```

Both proposals of EPI [12, 13] assume that $\mathbb{E}C(\rho_1(x))$ can be computed without error for every portfolio x. Reference [12] notes that $\mathbb{E}C(\rho_1(x))$ could be computed via simulation, but does not offer a sampling routine or convergence results. Since $\mathbb{E}C(\rho_1(x))$ can only be approximated via Monte Carlo simulations in this application, this section will provide a modification of Algorithm 2 to handle this situation and provides a proof of convergence for the resulting algorithm.

EPI is modified to manage Monte Carlo estimates by the inclusion of a sampling schedule, $(\xi_n)_{n\geq 0}$. To make the algorithm converge, the number of samples for each portfolio in iteration n must be $\mathcal{O}(n^\gamma)$, where $\gamma > 2$. Lines 12–14 of Algorithm 2 are replaced by those described in Algorithm 3.

Algorithm 3: Sampling routine to replace lines 12 − 14 of Algorithm 2

```
for $j = 1$ to $\xi_n$ do
    Get an outcome $\omega_j^n$
    Set $v_1^n = v_2^n = \cdots = v_{k+1}^n = 0$
    for $i = 1$ to $k + 1$ do
        $v_i^n = \frac{j-1}{j} v_i^n + \frac{1}{j} C(\rho_1(\pi_i^n, \omega_j^n))$
    end for
end for
```

Theorem 2 *Algorithm 2 when modified by Algorithm 3 converges to the optimal portfolio when the cost function C is bounded and the sampling sequence $(\xi_n)_{n\geq 0}$ increases with $\mathcal{O}(n^\gamma)$, with $\gamma > 2$.*

This version of the algorithm was developed as a part of this research, but its proof of convergence is beyond the scope of this paper, and is instead given in [32].

7 Algorithm behavior and numerical results

Although SGPO is guaranteed to converge in some situations, it may not in all situations. Often, it gets trapped in a local minimum instead of a global minimum. This tends to happen when there are multiple local minima that cannot reach each other by changing one or two projects. In other situations, the algorithm spends time cycling between portfolios, with portfolio changes occurring less often as the iteration number increases. Portfolio changes are shown by the dots in the lower graph of Fig. 4. In later iterations, however, SGPO tends to cycle through nearly equally good portfolios. Portfolio changes are still happening in the later SGPO iteration of Fig. 5, but the estimated value remains stable.

As a side note, trial runs of the algorithm show that marginal values can be extremely portfolio-dependent; this is shown in Fig. 4. This is to be expected, since funding two competing projects will diminish the marginal values of each. Such effects have been observed in actual R&D values [28].

The performance of SGPO was compared to some of its main competitors: a simulated annealing (SA) algorithm proposed by [31] and a Monte Carlo-based evolutionary algorithm, proposed by [13], as modified in Sect. 6. Unlike SGPO, both SA

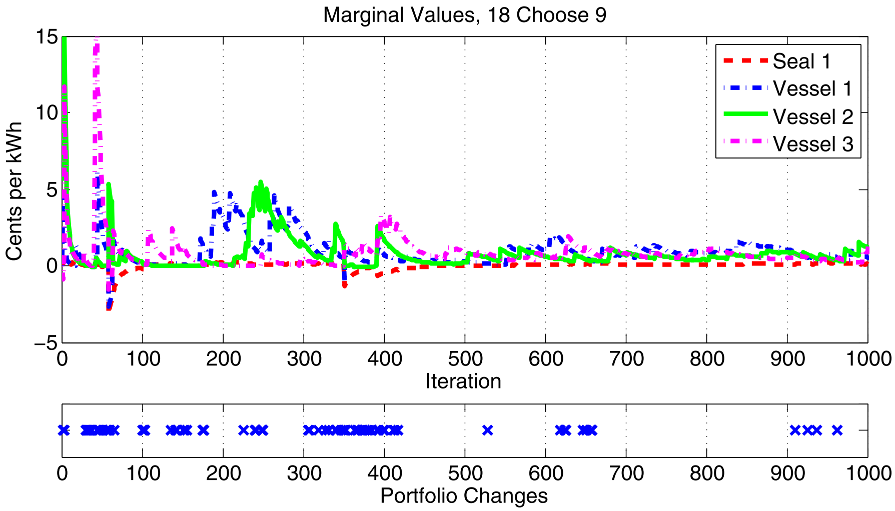

Fig. 4 Marginal values when all project costs are equal; 9 of the 18 projects are chosen. *Dots in the lower graph* signify portfolio changes. Marginal values can change drastically with the portfolio composition. Portfolios change less rapidly as the number of iterations increases, but do not necessarily settle on a single portfolio

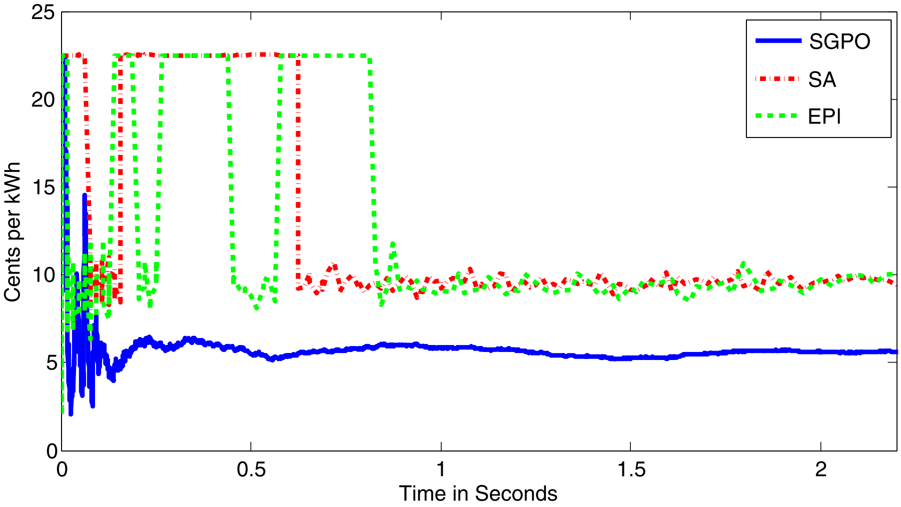

Fig. 5 Sample paths for SGPO, SA and EPI. The estimated value for the selection, $\bar{V}^n$, is plotted against the run time in seconds. Termination for all algorithms is roughly equivalent to 5,000 SGPO iterations. Note that in early iterations with some frequency both SA and EPI choose the poorer portfolio from the set of portfolios to be compared

and EPI are provably convergent to the global optimum with probability one for all cost functions. The experimental question that remains to be resolved is the rate of convergence and the reliability of each algorithm in terms of solution quality within a specific computing budget.

Both SA and EPI used a sampling size of n^{γ}, where $\gamma = 2.00001$, which was rounded to the nearest unit. Each iteration of EPI had four portfolios besides the selection from the previous iteration. Neighborhoods in the EPI algorithm around a portfolio x were designated as the set of all feasible portfolios where at most two portfolio items differed. For example, if $x_i = 1$ and $x_j = 0$, then the portfolio y, where $y_i = 0$ and $y_j = 1$, $y_k = x_k$, and $k \neq i, j$ would be in the neighborhood of x as long as y is feasible.

Choosing competing portfolios in non-equal cost cases is done is done in a slightly different manner. Whenever the algorithm calls for a completely random portfolio, a vector of random contributions for each project, $(\hat{d}_1, \dots, \hat{d}_m)$ is generated. Then, the following knapsack problem is solved to generate the portfolio $\hat{x}$,

$$\begin{aligned} \tilde{x} = \arg\max \sum_{i=1}^{m} \hat{d}_i x_i \\ \text{subject to} \sum_{i=1}^{m} c_i x_i \leq b, \\ x_i \in \{0, 1\}. \end{aligned}$$

Whenever the algorithm calls for a random portfolio within a neighborhood, a project currently in the portfolio is selected for removal. Let $i_1, \dots, i_{m(in)}$ be the indices of the remaining projects within the portfolio and $j_1, \dots, j_{m(out)}$ be the indices of the projects that were never in the portfolio. A vector of random costs, $(\hat{d}_{i_1}, \dots, \hat{d}_{i_{m(out)}})$, is generated for all the projects that were never in the portfolio. A portfolio of additional portfolio projects $\tilde{y}$ is found by solving the following knapsack program,

$$\begin{aligned} \tilde{y} = \arg\max \sum_{k=j_1}^{j_{m(out)}} \hat{d}_k x_k \\ \text{subject to} \sum_{k=j_1}^{j_{m(out)}} c_k x_k \leq b - \sum_{\ell=i_1}^{i_{m(in)}} c_\ell, \\ x_k \in \{0, 1\}. \end{aligned}$$

The additional projects selected are added to the projects of the original portfolio left after the removal of the random project to produce the new portfolio $\tilde{x}$.

Figure 5 is a collection of estimated portfolio values as a function of cost function evaluations. SGPO tends to a reasonable portfolio quickly, while SA and EPI often take a greater number of iterations to reach a low-cost portfolio. For a particular problem SGPO moves quickly to a low-cost portfolio, but it may not quickly account for high-cost, low-probability events. These are represented by the vertical jumps of the SGPO line in the figure. In contrast, both SA and EPI choose a portfolio after sampling it many times, and tends therefore to avoid portfolios with such events.

Figures 6 and 7 show the empirical probability density function of each method. Each method was run for 1000 iterations to obtain a portfolio selection. The value of

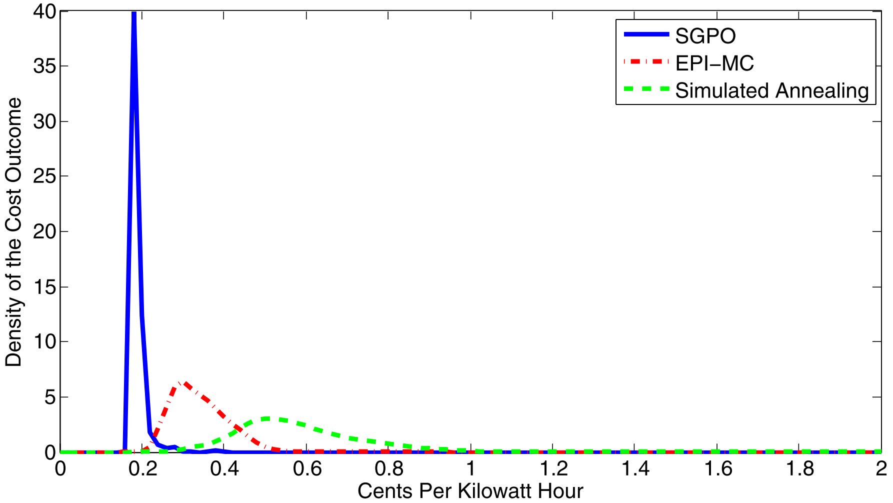

Fig. 6 Empirical PDFs for SGPO, EPI and SA when 20 projects are chosen out of 100. All algorithms are given the computational equivalent of 100 SGPO iterations. The number of possible portfolios for this problem is 5.3598 $\times 10^{20}$. The graph is a restriction of the outputs to those that are under 2 cents per kWh. Some outcomes for simulated annealing had expected costs of more than 30 cents per kilowatt hour

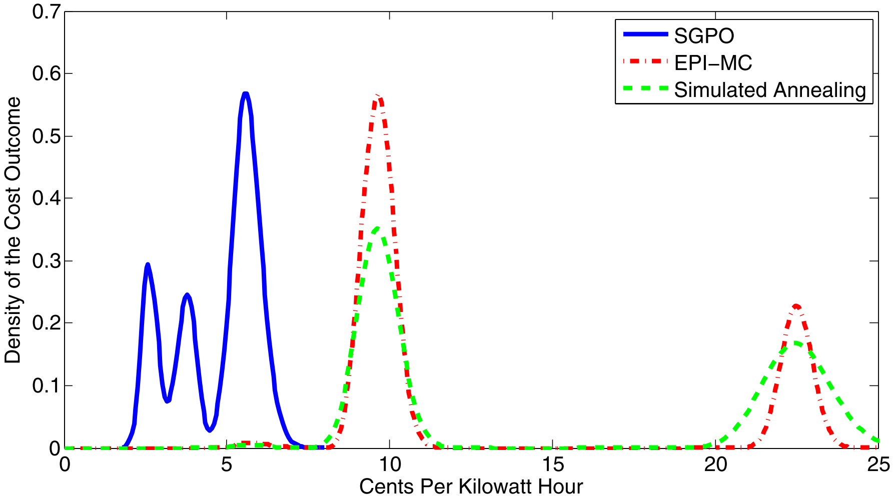

Fig. 7 Empirical PDFs for SGPO, EPI and SA when there are 15 projects with unequal costs that range from 1 to 8 with an average of 2.867 and a maximum portfolio cost of 10. The graph is truncated to 12 cents per kilowatt hour. Both simulated annealing and EPI had significant density around 22.5 cents per hour. There were 3,000 SGPO iterations to determine each portfolio; time-equivalent numbers of iterations for SA and SGPO were used for comparison

each portfolio selected in cents per kilowatt hour was estimated through 500 Monte Carlo simulations. The empirical probability density function $\hat{p}(x)$ for each method was then determined by discretizing the cost axis, counting the number of outputs

Table 2 Mean, standard deviation, minimum and maximum for each algorithm in cents per kWh. SGPO consistently outperformed its competitors in most metrics

¢/ kWh	18 Choose 9, 100 It.			18 Choose 9, 1,000 It.			18 Choose 9, 5,000 It.		
	SGPO	EPI	SA	SGPO	EPI	SA	SGPO	EPI	SA
Mean	0.78	1.66	6.43	0.62	1.31	4.92	0.61	1.16	3.93
Std.	0.45	1.03	10.46	0.12	0.50	7.00	0.04	0.51	4.67
Min	0.49	0.57	0.58	0.50	0.57	0.63	0.50	0.53	0.67
Max	7.41	7.05	47.70	3.43	5.69	48.03	0.79	6.46	45.70
¢/ kWh	100 Choose 20, 100 It.			Knapsack 15, 5000 It.			Knapsack 30, 3000 It.		
	SGPO	EPI	SA	SGPO	EPI	SA	SGPO	EPI	SA
Mean	0.34	0.63	3.22	4.72	13.20	15.28	1.88	14.57	16.68
Std.	0.41	0.31	5.63	1.27	5.82	6.42	1.27	7.12	7.06
Min	0.18	0.26	0.31	2.01	5.33	5.24	0.23	3.86	3.45
Max	4.71	4.05	31.29	7.02	22.51	22.51	5.47	22.50	22.51

in each bin, and then dividing by the total number of observations (1000) multiplied with the band width. That is for a bandwidth b,

$$\hat{p}(x) = \frac{\text{count of outputs in } [x - b/2, x + b/2)}{1000 * b}.$$

This approximates the probability density function of the true distribution of portfolio selections, $p(x)$.

SGPO was compared to EPI and SA in two problem settings. In all settings, the algorithms were run for a roughly equal amount of time. First, the algorithms were run in a setting where all problem costs are equal under various problem sizes. Second, the algorithms were run with unequal project costs. Table 2 gives statistics for the algorithm comparisons.

SGPO outperformed its competitors in all settings. The mean selection of SGPO was consistently better than the other two algorithms and it avoided the heavy tails that were demonstrated particularly by SA. Additionally, the given problems may not have demonstrated the full strengths of SGPO. It should greatly outperform SA and EPI on very large decision spaces since the latter two algorithms rely on random sampling to some extent. SA and EPI get to sample relatively few portfolios due to the increasing number of samples required each iteration for convergence. It is likely that even in the largest test, where 20 projects were chosen out of 100, there was a significant proportion of acceptable portfolios. This would explain why SA was still within an order of magnitude in this setting, even though it relies almost exclusively on a random search.

8 Discussion

This paper presents a flexible model for the one-stage R&D portfolio problem in Sect. 3. It allows technical interactions between projects to be modeled in a logical manner with a minimal number of parameters that need to be specified. Section 5 presented a new algorithm, SPGO, to find a good solution to the R&D portfolio optimization problem quickly, and proved convergence in some situations. Although not provably convergent in all situations, it had better performance than its competitors, EPI and SA, in a test problem involving Solid Oxide Fuel Cells. When the cost function can only be evaluated a small number of times, SGPO may be a better solution method than current stochastic optimization algorithms. More work is needed to determine the range of problems for which SGPO is and is not convergent. Nevertheless, the provably convergent algorithms may be useful in one-stage R&D portfolio optimization where computing time is not limited.

References

1. Alrefaei, M.H., Andradottir, S.: A simulated annealing algorithm with constant temperature for discrete stochastic optimization. Manag. Sci. **45**(5), 748–764 (1999)
2. Alrefaei, M.H., Andradottir, S.: A modification of the stochastic ruler method for discrete stochastic optimization. Eur. J. Oper. Res. **133**(1), 160–182 (2001)
3. Alrefaei, M.H., Andradottir, S.: Discrete stochastic optimization using variants of the stochastic ruler method. Nav. Res. Logist. **52**(4), 344–360 (2005)
4. Andradottir, S.: A method for discrete stochastic optimization. Manag. Sci. **41**(12), 1946–1961 (1995)
5. Andradottir, S.: Accelerating the convergence of random search methods for discrete stochastic optimization. ACM Trans. Model. Comput. Simul. **9**(4), 349–380 (1999)
6. Asher, D.T.: A linear programming model for the allocation of R&D efforts. IRE Trans. Eng. Manag. **9**, 154–157 (1962)
7. Bard, J.F.: Parallel funding of R&D tasks with probabilistic outcomes. Manag. Sci. **31**(7), 814–828 (1985)
8. Beaujon, G.J., Marin, S.P., McDonald, G.C.: Balancing and optimizing a portfolio of R&D projects. J. Oper. Res. Soc. **48**(1), 18–40 (2001)
9. Bianchi, L., Dorigo, M., Gambardella, L.M., Gutjahr, W.J.: Metaheuristics in stochastic combinatorial optimization: a survey. Technical report, Dalle Molle Institute for Artificial Intelligence (2006)
10. Blanning, R.W.: Variable-base budgeting for R&D. Manag. Sci. **27**(5), 547–558 (1981)
11. Cerny, V.: A thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm. J. Optim. Theory Appl. **45**, 41–51 (1985)
12. Chang, H.S., Fu, M.C., Hu, J., Marcus, S.I.: Simulation-Based Algorithms for Markov Decision Processes. Springer, Berlin (2007)
13. Chang, H.S., Lee, H.G., Fu, M.C., Marcus, S.I.: Evolutionary policy iteration for solving Markov decision processes. IEEE Trans. Autom. Control **50**, 1804–1808 (2005)
14. Charnes, A., Stedry, A.C.: A chance-constrained model for real-time control in research and development management. Manag. Sci. **12**(8), B353–B362 (1966)
15. Chen, C.H., He, D., Fu, M.: Efficient dynamic simulation allocation in ordinal optimization. IEEE Trans. Autom. Control **51**(12), 2005–2009 (2006)
16. Chen, C.H., Lin, J., Yücesan, E., Chick, S.E.: Simulation budget allocation for further enhancing the efficiency of ordinal optimization. Discrete Event Dyn. Syst. **10**(3), 251–270 (2000)
17. Chiu, L., Gear, T.E.: An application and case history of a dynamic R&D portfolio selection model. IEEE Trans. Eng. Manag. **26**, 2–7 (1979)
18. Coffin, M.A., Taylor, B.W.: Multiple criteria R&D project selection and scheduling using fuzzy logic. Comput. Oper. Res. **23**(3), 207–220 (1996)
19. Colvin, M., Maravelias, C.T.: A stochastic programming approach for clinical trial planning in new drug development. Comput. Chem. Eng., 2626–2642 (2007)
20. Doerner, K., Gutjahr, W.J., Hartl, R.F., Strauss, C., Stummer, C.: Pareto ant colony optimization: a metaheuristic approach to multiobjective portfolio selection. Ann. Oper. Res. **131**, 79–99 (2004)
21. Dorigo, M.: Optimization, learning and natural algorithms. PhD thesis, Dipartimento di Elettronica e Informazione, Politecnico di Milano, IT (1992)

22. Eilat, H., Golany, B., Shtub, A.: Constructing and evaluating balanced portfolios of R&D projects with interactions: a DEA based methodology. Eur. J. Oper. Res. **172**(3), 1018–1039 (2006)
23. Fang, Y., Chen, L., Fukushima, M.: A mixed R&D projects and securities portfolio selection model. Eur. J. Oper. Res. **185**(2), 700–715 (2008)
24. Fox, B.L., Heine, G.W.: Probabilistic search with overrides. Ann. Appl. Probab. **5**(4), 1087–1094 (1995)
25. Freeman, R.J.: A stochastic model for determining the size and allocation of the research budget. IRE Trans. Eng. Manag. **7**, 2–7 (1960)
26. Gelfand, S.B., Mitter, S.K.: Simulated annealing with noisy or imprecise energy measurements. J. Optim. Theory Appl. **62**(1), 49–62 (1989)
27. Gerencser, L., Hill, S.D., Vago, Z.: Optimization over discrete sets via SPSA. In: Farrington, P.A., Nembhard, H.B., Sturrock, D.T., Evans, G.W. (eds.): Winter Simulation Conference Proceedings of the 31st Conference on Winter Simulation: Simulation—A Bridge to the Future, pp. 466–470. ACM, New York (1999)
28. Girotra, K., Terwiesch, C., Ulrich, K.T.: Valuing R&D projects in a portfolio: Evidence from the pharmaceutical industry. Manag. Sci. **53**, 1452 (2007)
29. Goldberg, D.E., Segrest, P.: Finite Markov chain analysis of genetic algorithms. Proceedings of the Second International Conference on Genetic Algorithms and Their Application, pp. 1–8 (1987)
30. Gutjahr, W.J.: A converging aco algorithm for stochastic combinatorial optimization. In: Albrecht, A., Steinhoefl, K. (eds.) Stochastic Algorithms: Foundations and Applications. LNCS, vol. 2827, pp. 10–25. Springer, Berlin (2003)
31. Gutjahr, W.J., Pflug, G.C.: Simulated annealing for noisy cost functions. J. Glob. Optim. **8**, 1–13 (1996)
32. Hannah, L.A., Powell, W.B.: Proof of convergence for evolutionary policy iteration under a sampling regime. IEEE Trans. Autom. Control Lett. (2010, to appear)
33. Ho, Y.C., Sreenivas, R.S., Vakili, P.: Ordinal optimization of DEDS. Discrete Event Dyn. Syst. **2**(1), 61–88 (1992)
34. Holland, J.H.: Adaptation in Natural and Artificial Systems. University of Michigan Press, Ann Arbor (1975)
35. Hsu, J.C.: Multiple Comparisons: Theory and Methods. Chapman & Hall, London (1996)
36. Iyigun, M.G.: Multiobjective R and D portfolio analysis in highly uncertain environments. In: Technology Management: the New International Language, 27–31 October 1991, pp. 335–341 (1991)
37. Jin, Y., Branke, J.: Evolutionary optimization in uncertain environments—a survey. IEEE Trans. Evol. Comput. **9**, 303–317 (2005)
38. Keown, A.J.: Allocation of research and development funds: a zero-one goal programming approach. Omega **7**(4), 345–351 (1979)
39. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. Science **220**(4598), 671–680 (1983)
40. Klapka, J., Pinos, P.: Decision support system for multicriterial R&D and information systems projects selection. Eur. J. Oper. Res. **140**(2), 434–446 (2002)
41. Kramer, C.Y.: Extension of multiple range tests to group means with unequal numbers of replications. Biometrics **12**(3), 307–310 (1956)
42. Loch, C.H., Pich, M.T., Terwiesch, C., Urbschat, M.: Selecting R&D projects at BMW: a case study of adopting mathematical programming models. IEEE Trans. Eng. Manag. **48**(1), 70–80 (2001)
43. Lockett, A.G., Freeman, P.: Probabilistic networks and R&D portfolio selection. Oper. Res. Q. **21**(3), 353–359 (1970)
44. Medaglia, A.L., Graves, S.B., Ringuest, J.L.: A multiobjective evolutionary approach for linearly constrained project selection under uncertainty. Eur. J. Oper. Res. **179**, 869–894 (2007)
45. Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E.: Equations of state calculations by fast computing machines. J. Chem. Phys. **21**(6), 1087–1092 (1953)
46. Norkin, V.I., Ermoliev, Y.M., Ruszczynski, A.: On optimal allocation of indivisibles under uncertainty. Oper. Res. **46**(3), 381–395 (1998)
47. Prudius, A.A., Andradottir, S.: Two simulated annealing algorithms for noisy objective functions. In: Proceedings of the 2005 Winter Simulation Conference, pp. 797–802 (2005)
48. Ringuest, J.L., Graves, S.B.: The linear multi-objective R&D project selection problem. IEEE Trans. Eng. Manag. **36**(1), 54–57 (1989)
49. Ringuest, J.L., Graves, S.B.: The linear R&D project selection problem: an alternative to net present value. IEEE Trans. Eng. Manag. **37**(2), 143–146 (1990)

50. Ringuest, J.L., Graves, S.B., Case, R.H.: Mean-Gini analysis in R&D portfolio selection. Eur. J. Oper. Res. **154**, 157–169 (2004)
51. Rogers, M.J., Gupta, A., Maranas, C.D.: Real options based analysis of optimal pharmaceutical research and development portfolios. Ind. Eng. Chem. Res. **41**(25), 6607–6620 (2002)
52. Rudolph, G.: Convergence analysis of canonical genetic algorithms. IEEE Trans. Neural Netw. **5**(1), 96–101 (1994)
53. Rudolph, G.: Finite Markov chain results in evolutionary computation: a tour d'horizon. Fundam. Inform. **35**, 67–89 (1998)
54. Santhanam, R., Kyparisis, G.J.: A decision model for independent information selection. Eur. J. Oper. Res. **89**(2), 380–399 (1996)
55. Schmidt, R.L.: A model for R&D selection with combined benefit, outcome and resource interactions. IEEE Trans. Eng. Manag. **40**(4), 403–410 (1993)
56. Shi, L., Olafsson, S.: Nested partitions method for stochastic optimization. Methodol. Comput. Appl. Probab. **2**(3), 271–291 (2000)
57. Souder, W.E.: Effectiveness of mathematical models for R&D project selection. Manag. Sci. **19**(8), 907–923 (1973)
58. Spall, J.C.: Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. IEEE Trans. Autom. Control **37**(3), 332–341 (1992)
59. Stummer, C., Heidenberger, K.: Interactive R&D portfolio analysis with project interdependencies and time profiles of multiple objectives. IEEE Trans. Eng. Manag. **50**(2), 175–183 (2003)
60. Talias, M.A.: Optimal decision indices for R&D project evaluation in the pharmaceutical industry: Pearson index versus Gittins index. Eur. J. Oper. Res. **177**(2), 1105–1112 (2007)
61. Taylor, B.W., Moore, L.J., Clayton, E.R.: R&D project selection and manpower allocation with integer nonlinear goal programming. Manag. Sci. **28**(10), 1149–1158 (1982)
62. Tukey, J.W.: The problem of multiple comparisons. Technical report, Princeton University (1953)
63. Wang, J., Hwang, W.L.: A fuzzy set approach for R&D portfolio selection using a real options valuation model. Omega **35**(3), 247–257 (2007)
64. Weingartner, H.M.: Capital budgeting of interrelates projects: survey and synthesis. Manag. Sci. **12**(7), 485–516 (1966)
65. Yan, D., Mukai, H.: Stochastic discrete optimization. SIAM J. Control Optim. **30**(3), 594–612 (1992)