

Optimal enough?

Manfred Gilli · Enrico Schumann

Received: 1 November 2009 / Revised: 15 March 2010 / Accepted: 21 June 2010 /
Published online: 14 July 2010
© Springer Science+Business Media, LLC 2010

Abstract An alleged weakness of heuristic optimisation methods is the stochastic character of their solutions: instead of finding the truly optimal solution, they only provide a stochastic approximation of this optimum. In this paper we look into a particular application, portfolio optimisation. We demonstrate that the randomness of the ‘optimal’ solution obtained from the algorithm can be made so small that for all practical purposes it can be neglected. More importantly, we look at the relevance of the remaining uncertainty in the out-of-sample period. The relationship between in-sample fit and out-of-sample performance is not monotonous, but still, we observe that up to a point better solutions in-sample lead to better solutions out-of-sample. Beyond this point there is no more cause for improving the solution any further: any in-sample improvement leads out-of-sample only to financially meaningless improvements and unpredictable changes (noise) in performance.

Keywords Optimisation heuristics · Portfolio optimisation · Threshold accepting

1 Introduction

The aim of portfolio selection is to derive decision rules that help investors to allocate their wealth. The best known of these rules, mean-variance selection (Markowitz

Both authors gratefully acknowledge financial support from the EU Commission through MRTN-CT-2006-034270 COMISEF; the data set was provided by DynaGest S.A., Geneva.

E. Schumann (✉)

Department of Econometrics, University of Geneva, Bd du Pont d’Arve 40, 1211 Geneva 4, Switzerland

e-mail: Enrico.Schumann@unige.ch

M. Gilli

Department of Econometrics and Swiss Finance Institute, University of Geneva, Bd du Pont d’Arve 40, 1211 Geneva 4, Switzerland

e-mail: Manfred.Gilli@unige.ch

1952), is often criticised for failing to take into account the non-Gaussian nature of financial time series. Hence alternative and—at least theoretically—superior models have been proposed, with selection criteria that take into account ‘fat tails’ or asymmetric return distributions (an example are partial moments, see Fishburn 1977). Unfortunately, many of these alternative models lead to optimisation problems that are much harder to solve, and are often infeasible for standard optimisation techniques like linear or quadratic programming. For several specific models, like Expected Shortfall (Rockafellar and Uryasev 2000), exact solutions are available, but these techniques are in general not flexible and often cannot accommodate even minor changes in the model; neither can they generally handle realistic constraints like cardinality restrictions or limits to transaction costs.

There is an alternative approach to solve such optimisation problems: heuristics. The term ‘heuristic’ is used in different scientific areas, with different but related meanings. Mathematicians use it to describe an explanation that is not provable, sometimes even incorrect, but leads to a correct conclusion nonetheless; in the language of psychologists, heuristics are rules of thumb for decision making that, though sometimes seemingly crude, work robustly in many circumstances (Gigerenzer 2004, 2008); in computer science heuristics have been described as methods or rules for decision making that are (i) simple, and (ii) give good results sufficiently often (Pearl 1984, p. 3). Winker and Maringer (2007), following Barr et al. (1995) and similarly Zanakis and Evans (1981), characterise the term ‘heuristic optimisation’ through several criteria:

- The method should produce ‘good’ stochastic approximations of the true optimum, where ‘good’ is measured in terms of solution quality and computing time.
- The method should be robust in case of comparatively small changes to the given problem, and also for changes in the parameter settings of the heuristic itself. Robustness, again, is measured in (changes in) solution quality and computing time.
- The technique should be easy to implement.
- Implementation and application of the technique should not require subjective elements.

For many techniques like Genetic Algorithms (Holland 1992) or Simulated Annealing (Kirkpatrick et al. 1983) a considerable theoretical background is available, including mathematical analysis of their convergence. More importantly, heuristics have been shown to work well for problems that are completely infeasible for classical optimisation approaches (Michalewicz and Fogel 2004; for applications in finance see Maringer 2005).

1.1 The stochastics of ‘optimal’ solutions

Heuristics are, with only few exceptions (like a standard Tabu Search), stochastic algorithms. Thus repeated runs, called restarts, of the optimisation algorithm will result in different solutions—even for identical starting points. It is this stochastic nature of the solutions that sometimes causes distrust, for how can we judge whether we have actually found a good solution?

When it comes to the stochastics of solutions, there are similarities and differences between heuristics and classical, deterministic methods. For a given problem with

multiple local minima—and very few practical problems are truly convex—repeated runs *with different starting points* will result in different solutions, for both types of techniques. If we characterise a solution (in our case, a portfolio) by its associated objective function value, we may consider the result of one restart as a realisation of a random variable with some unknown distribution \mathcal{F} . The shape of \mathcal{F} will depend on the chosen method, hence when we try to solve a given multi-modal problem with a deterministic method, the probability of obtaining, from one optimisation run, a solution of a given quality remains fixed (though unknown). Heuristics on the other hand can move away from local minima, hence allowing more iterations per restart generally changes the shape of \mathcal{F} . If the algorithm works properly, with an increasing number of iterations, the distribution becomes steeper (the solutions become less dispersed), and moves to the left, closer to the global minimum. In other words, with a heuristic, the probability for obtaining a solution of a given quality will also depend on the computational resources (iterations) spent on the problem.

In this paper we solve a portfolio optimisation problem with a heuristic technique called Threshold Accepting (Dueck and Scheuer 1990; Moscato and Fontanari 1990; Dueck and Winker 1992), a variant of Simulated Annealing. Our main argument does not relate to a specific technique, so we describe the algorithm only informally here. Threshold Accepting (TA) builds on a simple concept in optimisation called local search. A local search starts with a random feasible solution (that is, a random portfolio) which we call the ‘current solution’, representing the best solution we have so far. Then again randomly, a new solution close-by is chosen. ‘Close-by’ means that we slightly perturb the weights of the portfolio (e.g., we ‘sell’ a small quantity of one asset, and invest the proceeds in another asset). This new solution is called a neighbour. If it is better than the current solution, the new solution is accepted and becomes the current solution, if not, it is rejected, and another neighbour is selected. This procedure stops after a preset number of iterations.

Since a local search stops at the first local minimum encountered, TA makes a small adjustment to the procedure: when the algorithm evaluates a neighbour solution, it may also accept this new solution if it is worse than the current solution—as long as the impairment does not exceed a given threshold (hence the method’s name). This threshold is set rather generously initially, so that the algorithm may move freely in the search space. Over time, the thresholds are decreased; hence the algorithm gets more select and finally turns into a local search. For a detailed description of TA, see Winker (2001).

So, how to judge the quality of a solution obtained from a TA run? For some problems, the objective function can directly be interpreted. The simplest approach then is to look at the realised value of the objective function associated with the returned solution. Assume we minimise a portfolio’s Value-at-Risk, then we could compare our results with the Value-at-Risk of a benchmark portfolio. In principle, if we had an exact method available, we could then also judge how much we lose from using a heuristic. But there is an important empirical question to be asked: how is the in-sample ‘performance’ that our solution achieves related to its out-of-sample performance? Much research in portfolio optimisation relates to in-sample properties of different methods: given a data set, we aim to minimise drawdown, or ratios of losses to gains. But what we actually want to minimise is *future* drawdown,

or the ratio of *future* losses to *future* gains. There exists comparatively little research into how these objectives relate to the quantities that we actually optimise. In other words, when we evaluate the quality of a solution, we need to look at its out-of-sample performance.¹

The point that we stress in this paper can be summarised as follows: the aim of portfolio selection is to find decision rules that tell investors how to invest. To this purpose, we set up a model (a selection criterion and constraints), but this model will depend on future asset prices which we do not know. Hence, we need to approximate/forecast/estimate/simulate these prices, and then solve the model. Zanakis and Evans (1981, p. 85) list cases where a heuristic should be the method of choice, their first case being ‘Inexact or limited data used to estimate model parameters [that] may inherently contain errors much larger than the “suboptimality” of a good heuristic.’ We shall argue that portfolio selection belongs to this class of optimisation problems.

We show that a heuristic technique can give a good solution to the model. Better solutions in-sample (i.e., solutions of the model) also lead to better solutions out-of-sample (the actual problem). But this holds true only up to a certain point: beyond this point, further improvement in-sample does not lead to meaningful improvement out-of-sample any more. If our in-sample solution is ‘good enough’, there is no more need to find a better one. The remainder of this paper is structured as follows: in Sect. 2 we briefly describe the optimisation problem and our data. Sections 3 and 4 then discuss both the in-sample and out-of-sample performance of our model. Section 5 concludes.

2 Data and methodology

There are n_A risky assets available, with current prices collected in a vector p_0 . We are endowed with an initial wealth v_0 , and wish to select a portfolio $x = [x_1 \ x_2 \ \dots \ x_{n_A}]'$ of the given assets such that

$$v_0 = x' p_0.$$

The vector x stores the number of shares or contracts, that is, integer numbers. The chosen portfolio is held for one period, from now (time 0) to time T . End-of-period wealth is given by

$$v_T = x' p_T,$$

where the vector p_T holds the asset prices at T . Since these prices are not known at the time when the portfolio is formed, v_T will be a random variable, following some

¹This also holds for optimisation strategies that aim to incorporate estimation uncertainty into the procedure, so notably robust optimisation approaches. If a robust optimisation model can only be solved by a stochastic method, we should again evaluate the stochastics of the solution in relation to out-of-sample performance; we need to check whether we have a meaningful, exploitable relationship between in-sample and out-of-sample quality.

unknown distribution. It is often convenient to rescale v_T to a return r_T , that is

$$r_T = \frac{v_T}{v_0} - 1.$$

Let \mathcal{J} be the set of assets in the portfolio, then our optimisation problem can be written as

$$\begin{aligned} \min_x \Phi, \\ x_j^{\text{inf}} \leq x_j \leq x_j^{\text{sup}} \quad j \in \mathcal{J}, \\ K_{\text{inf}} \leq \#\{\mathcal{J}\} \leq K_{\text{sup}}. \end{aligned}$$

We use downside semi-variance, or more precisely the lower partial moment of order two, as our objective function Φ . Semi-variance is defined as $E(\max(0, r_d - r_T)^2)$, where E is the expectations operator and r_d is a desired-return threshold. (Later on, we will always set r_d to zero.) For a sample of portfolio returns $r_T^{(1)}, r_T^{(2)}, \dots, r_T^{(n_S)}$, the semi-variance can be computed as

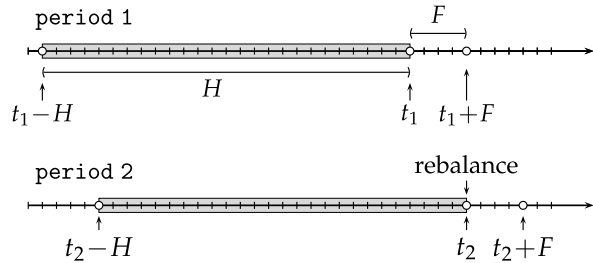
$$\frac{1}{n_S} \sum_{s=1}^{n_S} \max(0, r_d - r_T^{(s)})^2,$$

where n_S stands for the number of observations, or scenarios. x_j^{inf} and x_j^{sup} are minimum and maximum holding sizes, respectively, for those assets included in the portfolio (i.e., those in \mathcal{J}). K_{inf} and K_{sup} are cardinality constraints which set a minimum and maximum number of assets in \mathcal{J} . We set $x^{\text{inf}} = 1\%$ and $x^{\text{sup}} = 5\%$, thus we do not allow short positions. An upper cardinality is set to 50; a lower cardinality of 20 is implicitly defined by x^{sup} . We do not include a riskless asset. Since our algorithm works with actual position sizes, that is integer numbers, a small fraction of less than 1% of the portfolio is usually left uninvested.

We do not include a minimum-return constraint, but solely minimise the risk of the portfolio (defined as the semi-variance). This is equivalent to assuming equal means for all assets. There are several reasons that justify such a model. Firstly, individual future stock returns are so difficult to predict that the cost of an equal-mean assumption is often lower than the cost of incorrectly measured means, see Brandt (2009) for an overview of analytic and empirical results. Secondly, if we cannot really control the reward-side of the portfolio, then lowering portfolio risk may still lead to better risk-adjusted performance (Chan et al. 1999; Clarke et al. 2006). Furthermore, there is evidence that low-risk stocks actually yield higher returns than justified by pricing models (Blitz and van Vliet 2007). In any case, a low-risk portfolio is still a risky portfolio and should thus command a risk premium. Altogether there is strong empirical evidence that such an approach leads to portfolios that perform well in out-of-sample tests (see Board and Sutcliffe 1994, Chan et al. 1999 for variance-minimisation, Gilli and Schumann (forthcoming) for alternative risk functions).

The data set comprises more than 500 price series of European companies from the Dow Jones STOXX universe, at a daily frequency, spanning the period from March 2000 to March 2008. All stocks are denominated in euro.

Fig. 1 Moving-window optimisation



2.1 Moving windows

We conduct rolling-window backtests with an historical window of length H , and an out-of-sample holding period of length F . We set H to 250 business days, F to 63 business days. Thus we optimise at point in time t_1 on data from $t_1 - H$ to $t_1 - 1$, the resulting portfolio is held until $t_2 = t_1 + F$. At this point, a new optimal portfolio is computed, using data from $t_2 - H$ until $t_2 - 1$, and the existing portfolio is rebalanced. This new portfolio is then held until $t_3 = t_2 + F$, and so on. This is illustrated in Fig. 1 for the first two periods. With our data set, we have 35 investment periods. We optimise the first time on 06 March 2000 (t_1), the last date is 31 December 2007 (t_{35}). All periods are summarised in Table 1.

To solve our optimisation model, we need to obtain a sample of returns $r_T^{(1)}, r_T^{(2)}, \dots, r_T^{(n_S)}$ for a given period. The simplest approach is to use historic returns as scenarios, i.e., to treat them as a sample of realisations of r_T ; but there is evidence that the empirical performance of portfolio models can be improved by alternative scenario generation methods. Therefore, from the historical window in every period we create scenarios via the bootstrapping procedure described in Gilli and Schumann (forthcoming); the out-of-sample data for every period are the actual, historical data. A thorough discussion of the scenario generation technique is beyond the scope of this paper, and it is not necessary here: what is important is that the scenarios are created once and then saved, so all optimisation runs are conducted on the same data set. In other words, any variation in the results of repeated optimisation runs must arise from the fact that the optimisation technique is stochastic.

3 In-sample results

The portfolio optimisation is handled with Threshold Accepting (TA; see Gilli and Schumann 2010, for more details on the implementation for portfolio selection problems). We measure the computational resources that the optimisation algorithm employs by the number of iterations (i.e., the total number of objective function evaluations). With an increasing number of iterations the average quality of a solution per restart should improve while the solutions' variability should decrease; eventually the solutions' distribution \mathcal{F} should collapse into a single point (the global minimum) as

Table 1 Investment periods: rank correlations between in-sample fit and out-of-sample risk and return. The in-sample dates correspond to $t_{\text{period}} - H$ and $t_{\text{period}} - 1$ in Fig. 1; the out-of-sample dates to t_{period} and $t_{\text{period}} + F$

Period	In-sample period (dd.mm.yyyy)	Out-of-sample period (dd.mm.yyyy)	Rank correlation in-sample fit vs out-of-sample ...	
			...risk	...return
1	05.03.1999–06.03.2000	07.03.2000–26.05.2000	−0.53	0.70
2	27.05.1999–26.05.2000	29.05.2000–21.08.2000	0.51	0.25
3	20.08.1999–21.08.2000	22.08.2000–13.11.2000	0.77	−0.70
4	12.11.1999–13.11.2000	14.11.2000–05.02.2001	0.59	0.07
5	04.02.2000–05.02.2001	06.02.2001–30.04.2001	0.88	−0.25
6	28.04.2000–30.04.2001	01.05.2001–23.07.2001	0.79	−0.65
7	21.07.2000–23.07.2001	24.07.2001–15.10.2001	0.92	−0.83
8	13.10.2000–15.10.2001	16.10.2001–07.01.2002	0.91	0.71
9	05.01.2001–07.01.2002	08.01.2002–01.04.2002	0.88	0.13
10	30.03.2001–01.04.2002	02.04.2002–24.06.2002	0.91	−0.88
11	22.06.2001–24.06.2002	25.06.2002–16.09.2002	0.96	−0.58
12	10.09.2001–16.09.2002	17.09.2002–09.12.2002	0.97	0.73
13	07.12.2001–09.12.2002	10.12.2002–03.03.2003	0.98	−0.93
14	01.03.2002–03.03.2003	04.03.2003–23.05.2003	0.97	0.29
15	23.05.2002–23.05.2003	26.05.2003–18.08.2003	0.95	0.72
16	16.08.2002–18.08.2003	19.08.2003–10.11.2003	0.94	0.76
17	08.11.2002–10.11.2003	11.11.2003–02.02.2004	0.87	0.22
18	31.01.2003–02.02.2004	03.02.2004–26.04.2004	0.95	−0.74
19	25.04.2003–26.04.2004	27.04.2004–19.07.2004	0.86	−0.90
20	18.07.2003–19.07.2004	20.07.2004–11.10.2004	0.95	−0.30
21	10.10.2003–11.10.2004	12.10.2004–03.01.2005	0.92	−0.79
22	02.01.2004–03.01.2005	04.01.2005–28.03.2005	0.68	0.23
23	26.03.2004–28.03.2005	29.03.2005–20.06.2005	0.87	−0.63
24	18.06.2004–20.06.2005	21.06.2005–12.09.2005	0.90	−0.11
25	10.09.2004–12.09.2005	13.09.2005–05.12.2005	0.52	0.55
26	03.12.2004–05.12.2005	06.12.2005–27.02.2006	0.86	−0.36
27	25.02.2005–27.02.2006	28.02.2006–22.05.2006	0.12	−0.16
28	20.05.2005–22.05.2006	23.05.2006–14.08.2006	0.89	−0.39
29	12.08.2005–14.08.2006	15.08.2006–06.11.2006	0.93	−0.44
30	04.11.2005–06.11.2006	07.11.2006–29.01.2007	0.66	−0.19
31	27.01.2006–29.01.2007	30.01.2007–23.04.2007	0.89	0.85
32	21.04.2006–23.04.2007	24.04.2007–16.07.2007	0.12	0.86
33	14.07.2006–16.07.2007	17.07.2007–08.10.2007	0.84	0.32
34	06.10.2006–08.10.2007	09.10.2007–31.12.2007	0.92	−0.87
35	29.12.2006–31.12.2007	01.01.2008–24.03.2008	0.94	−0.37
Average			0.77	−0.11

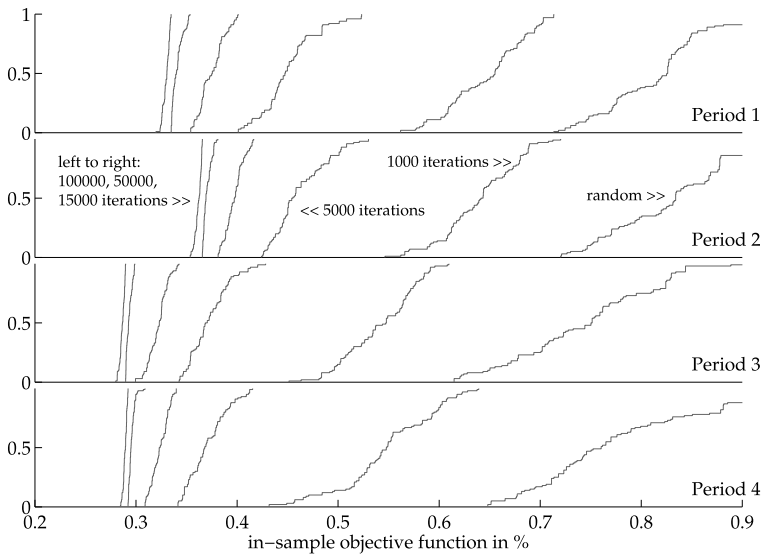


Fig. 2 In-sample convergence: estimates of the distribution \mathcal{F} with different numbers of iterations

the number of iterations goes to infinity.² In other words, if we run a TA with I_1 iterations, and one with I_2 iterations, where $I_1 < I_2$, then on average the TA with I_2 iterations will give better solutions. In-sample, thus, we face a trade-off between solution quality and computational resources spent.

We test the performance of TA with 1, 1 000, 5 000, 15 000, 50 000 and 100 000 iterations. With only 1 iteration, we practically obtain random (but feasible) portfolios. Hence we will write ‘random portfolios’ instead of ‘1 iteration’. Setting the computational resources fixes the distributions \mathcal{F} (as described in the Introduction) from which we draw our solutions. To approximate these distributions, we run the algorithm 100 times for every level of iterations, and compute the empirical cumulative distributions functions as estimates of the true distributions. Figure 2 shows the empirical results for the first four periods; for later periods we obtain similar results. In all figures we take the square root of the objective function values in order to obtain interpretable numbers.

With an increasing number of iterations the distributions become rapidly steeper and move to the left. There is little difference any more between the results for 50 000 and 100 000 iterations. Thus we can make the randomness of the in-sample objective function very small. To give a concrete example: in period 4 (the lower panel in Fig. 2) the realised objective function values range between 0.2845% and 0.2997% for 100 000 steps, while for our sample of random portfolios there is a range between 0.6031% and 0.9437%. (We generally give 4 digits to illustrate how small certain

²Winker and Maringer (2009) discuss the convergence of heuristics in the context of maximum likelihood estimation. Their analysis of convergence is also application-driven, that is, they investigate the convergence of a maximum likelihood estimator to the true parameter values in conjunction with the numerical convergence of the heuristic.

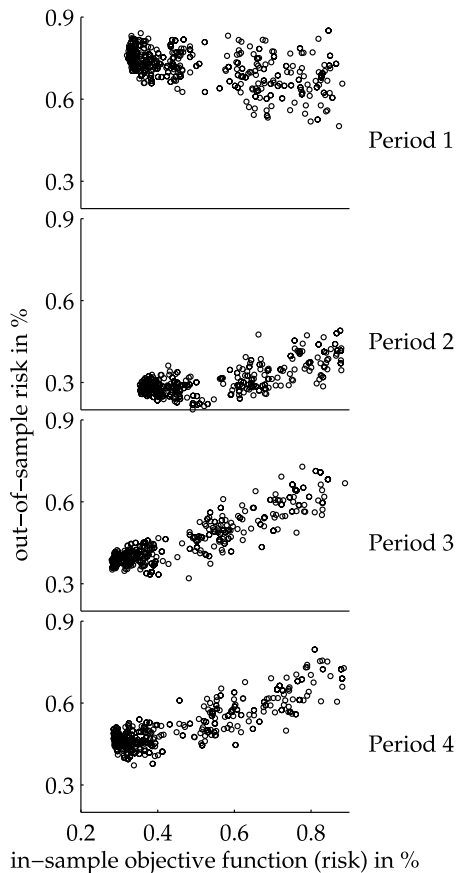
differences are, not because we think that working with such ‘precision’ is a good idea.) Of course, we have no guarantee that we have found the global minimum. But with a bound at zero, and a typical randomly-drawn portfolio having more than double the risk of our optimised portfolios, TA seems to consistently find ‘good’ solutions. In other words, TA seems well capable of solving our model.

4 Out-of-sample results

A given portfolio maps into an in-sample objective function value, but also into an out-of-sample risk (semi-variance) and return. This link between in-sample fit and actual out-of-sample performance is noisy, though: Fig. 3 shows a scatter plot of in-sample against out-of-sample risk for the first four periods (measured on a daily basis; again we take the square root of the semi-variance).

The picture shows characteristic features also present in the other periods. First of all, there is no monotonous relationship between in-sample fit and out-of-sample performance. For high in-sample objective function values (associated with the random portfolios), the out-of-sample results are widely scattered.

Fig. 3 In-sample risk (x -axis) and associated out-of-sample risk (y -axis)



the left, closer to the in-sample optimum, the cloud becomes denser. In particular, the leftmost part of the cloud, which covers only a narrow range in terms of the in-sample objective function, gives rise to comparatively more variable out-of-sample results. A concrete example: the objective function values of the 10 in-sample-best solutions for the third period pictured in Fig. 3 cover a range from 0.2798% to 0.2824%. We thus have a range of 0.0025%, or one-fourth of a basis point. The resulting out-of-sample risk associated with these solutions ranges between 0.3518% and 0.3875%, a range of 0.0357%, or three and a half basis points.

Altogether (and except for period one), the picture still is encouraging: lower in-sample risk leads to lower out-of-sample risk. Market conditions change, but on average when portfolio x is less risky than portfolio x' in-sample, it is also less risky out-of-sample. Table 1 gives the rank correlation between in-sample and out-of-sample risk for every subperiod. The average correlation coefficient is 0.77. So it seems that we can indeed control the risk of the portfolio.

But we do not want low-risk portfolios per se. We want a high risk-adjusted return, expressed for instance as a high ratio of return to volatility (the Sharpe ratio), or a high ratio of return to the square root of semi-variance (the Sortino ratio). Table 1 also gives the correlation between in-sample fit and out-of-sample return for all periods, but here the average correlation is only -0.11 , and generally correlations are varying strongly from one period to the next. So whatever the effect of portfolio optimisation on risk-adjusted performance, we need to check the longer term. For this purpose we compute the performance of 'rank-portfolios'. Rank-portfolios are constructed as follows. In every period, with iterations set to six different levels (1, 1 000, 5 000, 15 000, 50 000 and 100 000), and 100 restarts for each level, we have a total of 600 portfolios. We sort these portfolios according to their in-sample objective function value. A natural decision rule is to select the best in-sample portfolio, that is the portfolio with the lowest objective function value. We call this the rank-1 portfolio. Likewise, we can determine portfolios ranked from 2 to 600 for every period.

While the rank is determined in-sample, let us write r_t^i for the out-of-sample return of the portfolio with rank i in period t . The total out-of-sample return R^i along the path of the i th rank portfolio is

$$R^i = \prod_{t=1}^{35} (1 + r_t^i).$$

Hence R^1 is the total return for an investor who always chooses the best (rank-1) in-sample portfolio, R^2 is the total return for an investor who always chooses the second-best (rank-2) in-sample portfolio, and so on, until R^{600} gives the total return for the worst in-sample portfolios.

Figure 4 pictures the risk, the total return $R^i - 1$, the Sharpe ratio, and the Sortino ratio (all computed from daily data along the time paths and annualised) for all 600 rank portfolios, plotted against the respective rank. Portfolio risk has the clearest relationship with in-sample fit: it declines smoothly, but for the 200 best portfolios or so it is practically flat. The relation between rank and return is negative (as desired), but it is much more noisy, so it is less clear that improving a portfolio in-sample (i.e., advancing a rank) improves out-of-sample performance. Accordingly, the measures

Fig. 4 In-sample rank (lower is better) vs out-of-sample performance

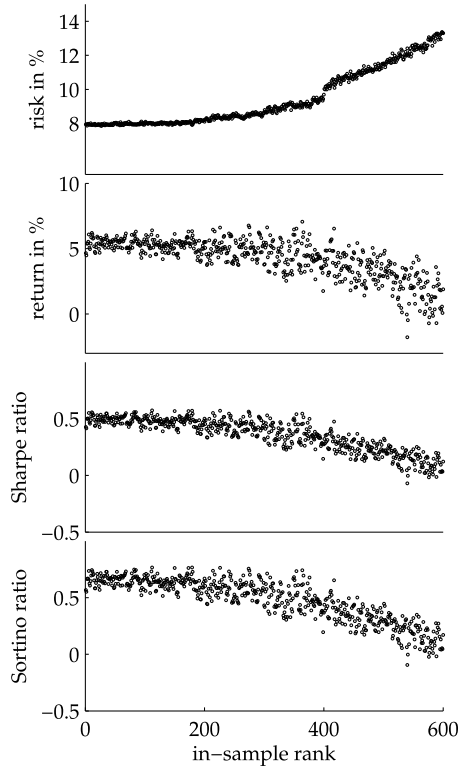


Table 2 Correlation between in-sample rank (lower is better) and out-of-sample risk, return, Sharpe ratio, and Sortino ratio

Rank correlation			Sharpe ratio	Sortino ratio
Ranks	Risk	Return		
all	0.99	-0.73	-0.87	-0.86
1-100	0.38	-0.02	-0.04	-0.04
101-200	0.57	-0.08	-0.14	-0.14
201-300	0.72	0.05	-0.03	-0.02
301-400	0.84	-0.13	-0.21	-0.20
401-500	0.95	-0.36	-0.47	-0.46
501-600	0.94	-0.49	-0.56	-0.55

of risk-adjusted return, which necessarily take into account returns, show a similarly noisy relation with in-sample rank.

To quantify these relations, we compute rank correlations between in-sample rank and out-of-sample performance. The results, given in Table 2, show that with decreasing ranks, the absolute correlations also decrease. Thus it becomes increasingly difficult to predict a portfolio’s out-of-sample quality from its in-sample fit. For the 100 best portfolios, the return and the return-risk ratios are hardly correlated any more

with the in-sample objective function, but there still appears to be a sizeable correlation of 0.38 between risk and in-sample fit that warrants investigation. For those 100 best portfolios the daily out-of-sample risk was on average 0.5029%, with a minimum of 0.4964% and a maximum of 0.5127%. Regressing risk on the ranks, we obtain a constant of 0.5050%, and a slope of -0.0000% . We regard this ‘improvement’ as zero. In other words, for the 100 in-sample-best portfolios, the associated out-of-sample risk cannot be meaningfully reduced by further improving the in-sample fit.

4.1 A portfolio contest

A correlation is actually a poor measure of financial relevance. As seen above, a high correlation may be obtained but still not signal a financially meaningful relationship. On the other hand, a low correlation does not necessarily imply that there is no exploitable relationship. We thus conduct a more direct test whether it is worth finding better portfolios. Given the relationship between in-sample and out-of-sample quality seen in Fig. 4—improving initially but then getting flat—we will consider returns here.

Assume there are two investors, A and B; both use TA to find portfolios. Investor A always lets his algorithm run for 100 000 iterations, while B uses only one iteration (i.e., random portfolios). Going back to Fig. 2, A will always pick portfolio from the leftmost distribution, while B will choose from the more dispersed random portfolios. In-sample, A’s portfolios will thus look much better than B’s, but how likely is A to outperform B? Given the noisy link in-sample to out-of-sample, we will unlikely observe that A’s portfolio dominates B’s in every period (not even to a higher order), but maybe it ‘almost dominates’ (Leshno and Levy 2002).

We start by randomly drawing one period from the 35 periods available. As explained before, each period can be split into an in-sample and an out-of-sample part. Then, for A, we randomly pick one portfolio out of the 100 that were optimised with 100 000 iterations on the in-sample part of this period, and record its out-of-sample performance r_1^A . We do the same for B, and hence obtain r_1^B . Next we pick again randomly one period (could even be the same period), choose again a portfolio for A and one for B, and record their out-of-sample returns, r_2^A and r_2^B . After p such draws, A’s normalised net worth will be

$$(1 + r_1^A)(1 + r_2^A)(1 + r_3^A) \cdots (1 + r_p^A),$$

and B’s will have grown to

$$(1 + r_1^B)(1 + r_2^B)(1 + r_3^B) \cdots (1 + r_p^B).$$

We then compute the geometric outperformance of A over B as

$$g = \frac{\prod_{i=1}^p (1 + r_i^A)}{\prod_{i=1}^p (1 + r_i^B)} - 1.$$

We repeat this exercise 10 000 times, for different levels of p . Thus, for a fixed investment horizon p , we obtain a distribution of g and can now compare how much better A’s portfolio performed compared with B’s.

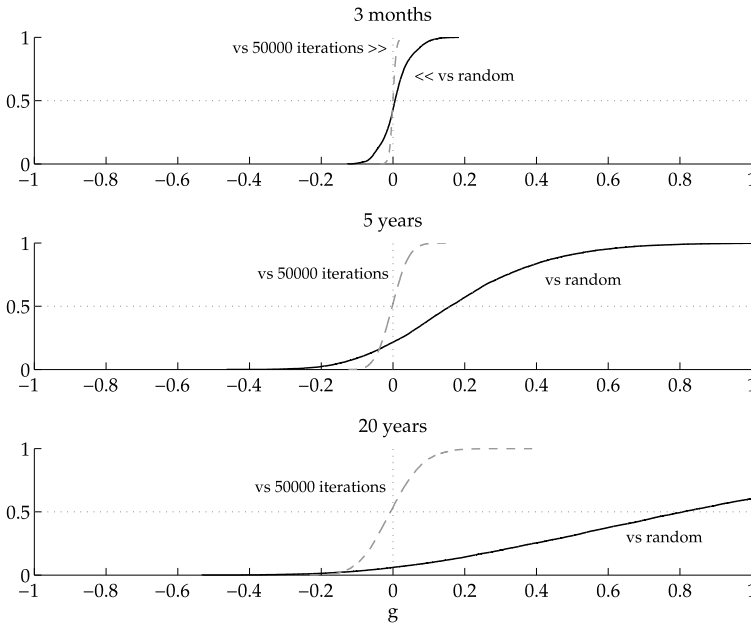


Fig. 5 Distribution of outperformance of portfolios with 100 000 iterations over portfolios with 50 000 iterations and random portfolios

Figure 5 shows these distributions of g . A value of $g = 1$ means that A’s wealth has grown to twice the wealth of B, while $g = 0$ indicates that A and B are equally wealthy. One period ($p = 1$) in our setting was three months, hence five years are 20 periods ($p = 20$), and twenty years are 80 periods ($p = 80$). There is only a probability of 57% that A outperforms B (i.e., a probability that $g > 0$) after one period of three months. After five years, the probability is 78%, and even after twenty years it is ‘only’ 94%. Note, however, that these distributions are not symmetric.

Now let B switch to a TA with 50 000 iterations, half the computational resources that A employs. Figure 5 gives a clear answer: even after twenty years, the distribution of g is symmetric around $g = 0$, hence there is little predictable difference between the portfolios of A and B.

We repeated this analysis with the Sortino ratio (results are available on request) with essentially identical results. In summary then, there seems little practical difference anymore between running an optimisation with 50 000 or 100 000 iterations.

5 Conclusion

In this paper we have analysed the stochastics of solutions obtained from a heuristic optimisation method when applied to a financial problem, portfolio optimisation. Our findings indicate that the uncertainty stemming from the optimisation technique can be made very (in fact, arbitrarily) small. The relationship between in-sample fit and out-of-sample performance is not monotonous, but still, on average, portfolios that

are better in-sample perform better out-of-sample. In essence, however, we show that finding ‘good’ portfolios suffices, with further in-sample improvements only leading to financially meaningless improvements (and unpredictable changes) out-of-sample.

This finding does not just apply to heuristics, but to ‘exact’ methods as well, with the difference that the latter techniques give the false impression of having provided the ‘optimal’ solution. The advantage of heuristics is that they give the analyst more freedom when setting up the optimisation model in the first place, since heuristics can accommodate alternative ways to model the data, or alternative objective functions. Thus, heuristics may be a valuable tool for truly improving the solution not of the model, but of the actual problem.

References

- Barr, R.S., Golden, B.L., Kelly, J.P., Resende, M.G.C., Stewart, W.R.: Designing and reporting on computational experiments with heuristic methods. *J. Heuristics* **1**(1), 9–32 (1995)
- Blitz, D.C., van Vliet, P.: The volatility effect. *J. Portf. Manag.* **34**(1), 120–113 (2007)
- Board, J.L.G., Sutcliffe, C.M.S.: Estimation methods in portfolio selection and the effectiveness of short sales restrictions: UK evidence. *Manag. Sci.* **40**(4), 516–534 (1994)
- Brandt, M.W.: Portfolio choice problems. In: Ait-Sahalia, Y., Hansen, L.P. (eds.) *Handbook of Financial Econometrics*. Elsevier, Amsterdam (2009)
- Chan, L.K.C., Karceski, J., Lakonishok, J.: On Portfolio Optimization: Forecasting Covariances and Choosing the Risk Model. *Rev. Financ. Stud.* **12**(5), 937–974 (1999)
- Clarke, R., de Silva, H., Thorley S.: Minimum-variance portfolios in the U.S. equity market. *J. Portf. Manag.* **33**(1), 10–24 (2006)
- Dueck, G., Scheuer, T.: Threshold accepting. A general purpose optimization algorithm superior to simulated annealing. *J. Comput. Phys.* **90**(1), 161–175 (1990)
- Dueck, G., Winker, P.: New concepts and algorithms for portfolio choice. *Appl. Stoch. Models Data Anal.* **8**(3), 159–178 (1992)
- Fishburn, P.C.: Mean-risk analysis with risk associated with below-target returns. *Am. Econ. Rev.* **67**(2), 116–126 (1977)
- Gigerenzer, G.: Fast and frugal heuristics: the tools of bounded rationality. In: Koehler, D.J., Harvey, N. (eds.) *Blackwell Handbook of Judgment and Decision Making*, Chap. 4, pp. 62–88. Blackwell, Oxford (2004)
- Gigerenzer, G.: Why heuristics work. *Perspect. Psychol. Sci.* **3**(1), 20–29 (2008)
- Gilli, M., Schumann, E.: Portfolio optimization with “threshold accepting”: a practical guide. In: Satchell, S.E. (ed.) *Optimizing Optimization: The Next Generation of Optimization Applications and Theory*. Elsevier, Amsterdam (2010)
- Gilli, M., Schumann, E.: Risk-reward optimisation for long-run investors: an empirical analysis. *Eur. Actuar. J.* (forthcoming). URL http://www.actuaries.org/Munich2009/Programme_EN.cfm
- Holland, J.H.: *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control and Artificial Intelligence*. MIT Press, Cambridge (1992)
- Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. *Science* **220**(4598), 671–680 (1983)
- Leshno, M., Levy, H.: Preferred by “All” and preferred by “Most” decision makers: almost stochastic dominance. *Manag. Sci.* **48**(8), 1074–1085 (2002)
- Maringer, D.: *Portfolio Management with Heuristic Optimization*. Springer, Berlin (2005)
- Markowitz, H.M.: Portfolio selection. *J. Finance* **7**(1), 77–91 (1952)
- Michalewicz, Z., Fogel, D.B.: *How to Solve It: Modern Heuristics*. Springer, Berlin (2004)
- Moscato, P., Fontanari, J.F.: Stochastic versus deterministic update in simulated annealing. *Phys. Lett. A* **146**(4), 204–208 (1990)
- Pearl, J.: *Heuristics*. Addison-Wesley, Reading (1984)
- Rockafellar, R.T., Uryasev, S.: Optimization of conditional value-at-risk. *J. Risk* **2**(3), 21–41 (2000)
- Winker, P.: *Optimization Heuristics in Econometrics: Applications of Threshold Accepting*. Wiley, New York (2001)

- Winker, P., Maringer, D.: The threshold accepting optimisation algorithm in economics and statistics. In: Kontoghiorghes, E.J., Gatu, C. (eds.) *Optimisation, Econometric and Financial Analysis. Advances in Computational Manag. Sci.*, vol. 9, pp. 107–125. Springer, Berlin (2007)
- Winker, P., Maringer, D.: The convergence of estimators based on heuristics: theory and application to a GARCH model. *Comput. Stat.* **24**(3), 533–550 (2009)
- Zanakis, S.H., Evans, J.R.: Heuristic “Optimization”: why, when, and how to use it. *Interfaces* **11**(5), 84–91 (1981)