Evolutionary Learning of Private Equity Recommitment Strategies

Emmanuel Kieffer*, Frederic Pinel*, Thomas Meyer§, Georges Gloukoviezoff ‡, Hakan Lucius‡ and Pascal Bouvry*†

*Faculty of Science, Technology and Medicine
University of Luxembourg
Email: firstname.name@uni.lu
†Interdisciplinary Centre for Security, Reliability and Trust
University of Luxembourg
Email: firstname.name@uni.lu
‡European Investment Bank
Luxembourg
Email: g.gloukoviezoff@eib.org,h.lucius@eib.org
§SimCorp Luxembourg SA
Luxembourg
Email: thomas.meyer@simcorp.com

Abstract—Achieving and maintaining high allocations to Private Equity and keeping allocations at the targeted level through recommitment strategies is a complex task which needs to be balanced against the risk of becoming a defaulting investor. When looking at recommitments we are quickly faced with a combinatorial explosion of the solution space, rendering explicit enumeration impossible. As a consequence, manual management if any is becoming time-consuming and error-prone. For this reason, investors need guidance and decision aid algorithms producing reliable, robust and trustworthy recommitment strategies. In this work, we propose to generate automatically recommitment strategies based on the evolution of symbolic expressions to provide clear and understandable decision rules to Private Equity experts and investors. To the best of our knowledge, this is the first time a methodology to learn recommitment strategies using evolutionary learning is proposed. Experiments demonstrate the capacity of the proposed approach to generate efficient and robust strategies, keeping a high degree of investment while bounding the risk of being overinvested.

Index Terms—Private Equity, Genetic Programming, Evolutionary Learning

I. INTRODUCTION

In recent years institutional investors – e.g. pension funds, insurers, endowments, development banks and sovereign wealth funds – have been building up sizable allocations to Private Equity (PE) and real assets (mainly infrastructure, real estate, forestry and farmland, energy and commodities). Successfully entering, managing, and exiting direct investments requires a high level of expertise, experience and staff incentives that most institutional investors are unable to provide. Therefore, their preferred modus operandi is investing indirectly as so-called limited partners (LPs) through limited partnership funds in which they commit a certain amount of capital for a given period of time. Commitments are binding and the related capital is progressively called over several years depending on the fund's management strategy. Unfor-tunately, capital calls cannot be known in advance leading to unallocated capital waiting to be called. To make matters worse, the committed capital is not entirely called before the fund starts distributing from divestments. These distributions occur when the fund is only few years old and can only be reinvested by committing to new PE funds. Generally, all the committed capital is not entirely drawn down which impacts the investor's exposure. Therefore, the capital actually invested (see [18]) is often thought to be the true exposure to PE. This is debatable as the LP's liquidity risk is the highest at the time of the commitment and during the fund's early years when the bulk of the capital is yet to be called but its investment level has not reached its peak. Investors need to draft a commitment-pacing strategy, i.e., on how to size and time their commitments in order to achieve and maintain the targeted allocation to PE while complying with the liquidity constraints imposed by the uncalled capital. So far portfolio models based on cash-flows tend to be over-simplistic and usually focus on commitment-pacing. Usually these models are spreadsheet-based, very simple and work through "trial-and-error". The impact of (alleged) skills when selecting high-quality funds is either not reflected or over-estimated. Often for the targeted portfolio composition there are no funds with desired characteristics available or not available at the time. Also, the interaction with the secondary market for funds is difficult to factor in, as they tend to arise precisely when LPs experience liquidity problems. The costs of maintaining the uncalled capital as dry powder is a widely overlooked expense of investing in PE [1] and still ignored by academic research [16]. In practice, LPs therefore run so-called overcommitment strategies, i.e., committing more capital in aggregate than actually available as dedicated resources, with the gap expected to be filled by future distributions from the existing portfolio of funds. Overcommitments share important commonalities

with leverage strategies and show similar rewards and risks, notably that of becoming a defaulting investor and incurring significant financial and reputational penalties. Achieving and maintaining high allocations to PE and keeping them at the targeted level is a complex task and needs to be balanced against the risk of becoming a defaulting investor. According to de Zwart in [8], this issue has received very little attention in the literature. Existing strategies are often naive and manually designed despite the fact that the costs of inefficiently (re)committing can be significant.

In this paper, we propose to go beyond handcrafted and over-simplistic commitment strategies using a simulation-based approach to discover new promising strategies. For this purpose, we consider an evolutionary learning approach based on the generation of symbolic expressions. These expressions are evolved with a Genetic Programming (GP) algorithm specially tailored for this task. The choice of symbolic expressions reflects the need for trustworthiness to have a correct understanding of their meaning. Although many learning approaches such as Deep learning have gained popularity, they remain a black box for many decision makers who struggle to put them into production especially in PE where data is somehow lacking.

The remainder of this paper is organized as follows. The next section provides a state of the art on existing recommitment strategies and describes related approaches based on symbolic expressions to tackle optimisation problems. Section III introduces formally the Private Equity Recommitment Problem (PERP). Section IV details the proposed methodology to learn recommitment strategies. Experiment setups and results are discussed in section V and VI. Finally, the last section provides our conclusions and proposes some possible perspectives.

II. RELATED WORKS

Recommitment strategies are essential to keep investor constantly invested at some target allocation. To the best of our knowledge, few studies have tried to model this as an optimisation problem. They generally rely on some rules of thumb lacking robustness and flexibility. In [5], authors considered that the entire PE allocation should be recommitted to new funds every year without taking into account past portfolios evolution. Nevin et al. in [17] based their recommitment strategy on average rates of distributions and commitments. New commitments should be made if the committed capital does not reach a target threshold to compensate the difference. This strategy assumes constants rates which seems very illusory over time. In [8], de Zwart et al. proposed recommitment strategies for funds aiming to maintain stable the exposure to PE. The strategy's key feature is the level of new commitments in a given period which depends on the current portfolio's characteristics. Importantly, de Zwart's strategies does not require to forecast funds'cashflows. Although they consider 100% PE portfolios, their last suggested strategy is a first attempt to design dynamic recommitment strategies relying on past portfolio development. Finally, Oberli et al. in

[18] extended de Zwart's work to multi-asset class portfolios including stocks and bonds.

These two last contributions solely rely on handcrafted recommitment strategies to control the investment degree (ID), i.e., PE exposure. While they are innovative and improving attempts without the need to forecast future cashflows, they have been built on specific and limited datasets with given market conditions. Building recommitment strategies in various market conditions is a challenging task. In this work, we investigate Evolutionary Learning to discover promising recommitment strategies modeled as symbolic expressions. This is the reason why we first discuss different domains where learning symbolic expressions have been very promising.

Learning symbolic expressions have been extensively considered for regression tasks [29] in which an interpretable model is fitted to data. Nevertheless, discovering symbolic expressions can be extended to solve general optimisation problems. Indeed, hyper-heuristics, i.e., a class of metaheuristics to generate heuristics, have been successfully employed to generate scoring functions for the Multi-dimensional Knapsack [10], [14]. Recent works [11], [15] focus on heuristic generation to provide a systematic and automatic way to generate directly heuristics components going from simple expressions to full blocks of code. The literature is replete with approaches relying on GP algorithms which evolve Abstract Syntax Trees (AST). The suitability of GP algorithms has been established by Fukunaga in [13] for the well-known SAT problem. They have the major advantage to automatize the assembly of the components required to create a heuristic. Recent variants such as Cartesian GP algorithms and Grammar-based GP algorithms are improvements of the classical GP algorithms. Cartesian GP algorithms is an alternative form that encodes a graph representation of a computer program. The Cartesian variant defines explicitly a size preventing bloat but can be very sensitive to parameters. In Grammar-based versions [2], [23], [24], a grammar in Backus-Naur Form (BNF) is considered to map linear genotypes to phenotype trees and have thus less structural difficulties. Heuristic generation encountered real successes in combinatorial optimization problems [21], [22] and more specifically in cutting and packing [4], scheduling [3] and other additional domains such as function optimization [20], real-time logistics [26].

Although their applications have been widely considered for discovering novel optimisation heuristics to NP-hard problems, GP algorithms and its variants have demonstrated capabilities in Evolutionary Learning. Wilson et al. in [28] have shown image processing aptitudes by evolving code with function sets dedicated for computer vision. Using a Cartesian GP variant, they confirmed that evolved controllers are competitive with state-of-the-art approaches for the Atari benchmark set and more important, they required less training time. Stock Price Prediction using Grammar Evolution have been successfully investigated in [9]. The authors observed that the effect of tweaking the grammar rules to provide different production options has a sensible impact on the prediction quality. Evolutionary Learning using the GP family of approaches have
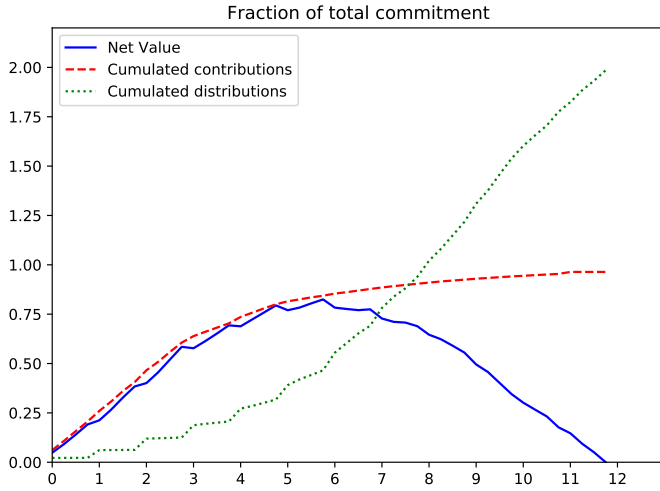
Fig. 1. Evolution of the Net Asset Value, cumulative distributions and contributions

been also naturally found in Automated Machine Learning (AutoML). Oslon et al. in [19] considered a GP algorithm to evolve a set of data transformations and Machine Learning models in their well-known tree-based pipeline optimisation tool (TPOT). Hyperparameters optimisation has been also a subject of investigation with the autostacker method developed by Chen et al. in [6].

III. PRIVATE EQUITY RECOMMITMENT PROBLEM (PERP)

The Private Equity Recommitment Problem is essentially a target allocation problem for an investor who decides to own a PE portfolio. Once capital has been committed to a fund, it is drawn down progressively during the *investment period* and the Net Asset Value (NAV) steadily increases until reaching a peak around the halfway point of the fund lifetime. PE funds have generally a lifetime ranging from 10 to 14 years. Once this peak has passed, the fund enters into a second phase called the *divestment period* in which the NAV starts decreasing due to distributions while capital calls strongly decline. Both periods are depicted in Fig. 1 for a single PE fund. One can observe that the maximum investment into the fund is made during a very short period of time. This is the reason why its is crucial to take advantage of the uncalled capital and possible early distributions by recommitting into new PE funds to counter-balance the *opportunity cost* and minimize underinvestment.

In order to quantify the true exposure, an investor should rely on the fraction of total allocated capital that is actually invested which can be defined as

$$ID_t = \frac{NAV_t}{NAV_t + Cash_t} \qquad (1)$$

where $NAV_t = \sum_{f}^{N(t)} NAV_t^f$ represents the NAV of all invested funds in the portfolio at period t. $Cash_t$ accounts for

the global uninvested cash in the portfolio, i.e., uncalled capital and distributions. Investors are looking for an investment degree approaching 1 from below by minimizing the dry powder ($Cash_t^f$) which should be ideally close to 0. The combination of multiple invested funds obtained with an intelligent recommitment pacing is a guarantee to maintain a high investment degree/exposure. Nevertheless, recommitments should be carefully estimated to avoid cash shortages, i.e., $Cash_t < 0$ or $ID_t > 1.0$. The latter happens only if the call requirements exceed the investor resource capacities. Becoming a defaulting investor once capital has been committed is subject to strong financial and reputational penalties. Being fully invested is thus a complex task for an investor which can be modeled as an optimisation problem. The latter referred to as PERP is a multi-period dynamic portfolio optimisation problem which is generally decomposed into a sequence of multiple single-period portfolio optimisation problems solved at each period t. Each single-period portfolio optimisation maximizes the investment degree at $t + 1$:

$$\min_{C_t} E_t \left[(1 - ID_{t+1})^2 \right] \qquad (2)$$

The optimal level of commitment at period t is therefore:

$$C_t = E_t \left(\frac{Cash_t + D_{t+1} - \sum_{i=1}^{\tau} \gamma_{t+1,i+1} C_{t-i}}{\gamma_{t+1,1}} \right) \qquad (3)$$

with C_t the commitment solution, E_t the conditional expectation, $Cash_t$ the uninvested cash in the portfolio, D_{t+1} representing distributions for the next period, C_{t-i} the capital committed i period ago and $\gamma_{t+1,i+1}$ is the fraction of the capital committed i periods ago. $\gamma_{t+1,i+1}$ permits to define the total capital call at the end of quarter $t + 1$, i.e., $CC_{t+1} = \sum_{i=0}^{\tau} \gamma_{t+1,i+1} C_{t-i}$ with τ representing the maximum fund age at which capital can still be called. Interested readers can refer to [8] for more details about the proof.

In order to compute the global optimal solution of this problem with this analytical solution, investors have to forecast the distributions as well as the capital called at the period $t+1$. Furthermore, $\gamma_{t+1,i+1}$ is determined only by the fund manager at $t + 1$. LP investors have no control on this factor. This is the reason why two approaches have been developed in the literature so far:

1) Cash flow predictions [7], [25]
2) Recommitment engineering using strategies [8], [18]

Cashflow predictions rely on direct forecasts of the cash inflows and outflows, .i.e., distributions and contributions respectively. The quality of the PERP analytical solution depends directly on the quality of the prediction. The unique advantage of cashflows prediction is the direct access to the analytical optimal solution. Its main drawback lies in the fact that prediction accuracy only occurs towards the end of the life cycle of funds. The second approach based on strategies is an indirect approach replacing the analytical solution by approximation rules which do not rely on any cashflow predictions. This last point is clearly a leverage when

funds are still at their infancy and cashflow predictions are not accurate. Nonetheless, discovering such strategies is a tedious and laborious task requiring multiple trials. In this work, we propose to automatise this task using evolutionary learning. The next section will now introduce the proposed methodology.

IV. LEARNING TO RECOMMIT

The benefit of generating approximation strategies is two-fold. First, they do not rely on any forecasting approaches and only exploit existing information. Then, they are more suitable for analysis by PE experts who can study them and therefore trust them. As aforementioned in the previous section, searching strategies require a lot of effort and need numerous simulations. As a consequence, we propose here-after to generate recommitment strategies with a GP algorithm in a similar manner than generative hyper-heuristics build heuristics for more classical optimisation problems.

A. Building artificial cashflows

In order to simulate portfolio's dynamic and evaluate recommitment strategies, one has to rely on cashflows data. These are historical funds' data that are argued to reflect the "true" behaviour of funds and thus capture the dynamics of PE and real assets best. Unfortunately, PE players protect their rich data histories. Some private market data providers generally sell data covering very specific periods and economies, but these data tend to be incomplete. Although, multiple works relied on these commercial libraries to train their model or simulate portfolio evolutions over time, we adopt another strategy to generate these cashflows.

The Yale University's endowment has been investing in PE since 1973. Their methodology for modelling illiquid asset funds as proposed by Takahashi and Alexander (see [25]) is often referred to as the *mother of all cashflows' models*. This Yale-model can be applied to PE and real asset funds, such as natural resources and infrastructure. The timing of all cashflows, as well as the return on the committed capital, is modelled as deterministic, i.e., in contrast to a probabilistic model, a single run of this model creates just one result for one set of input parameters and not a range of out-comes. Nevertheless, according to Takahashi and Alexander, the projections generated fit historical data surprisingly well. Instead of depending on a commercial solution to acquire historical cashflows which are often expensive and incomplete, synthetic fund cashflows have been preferred in this work as they represent a more practical solution. These synthetic fund cashflows are created by funnelling data generated by the robust and tried-and-tested, albeit over-simplistic, Yale-model through a noise-adding algorithm to construct a new dataset. The resulting dataset shows the statistical features and the useful patterns needed for capturing the liquidity risks associated with portfolio of funds. The synthetic cashflows implemented in this work have been provided by T.Meyer, an expert in PE and co-author of this paper.

B. Simulations and Strategies assessment

Strategies are mathematical expressions computing C_t at each recommitment period t. They can be evaluated by mean of portfolios simulations. As reported in [8], dynamic recommitment strategies assume pre-existing portfolios. Therefore, in this work, we build initial portfolios over one year by investing uniformly into 16 randomly selected PE funds. A single simulation consists in investing some capital to new funds every quarter. The amount of capital is determined by the strategy employed to recommit. Recommitment capital is then equally divided and assigned to four randomly selected new funds. Please note that we strictly follow and use the same parameters for simulations presented in [8] to fairly compare the authors' strategies with the ones obtained in this work. For that reason, 30 % overcommitment is applied in setting up all initial portfolios.

Recommitments are performed during T quarters starting with the first quarter of the second year. Simulations are stopped only when all underlying PE funds are totally divested. It is important to see if there is no delay effect when applying a specific strategy. In summary, active recommitments last T quarters but simulations keep running until no further contributions could lead to a cash shortage situation. Once simulations are over, the inherent investment degrees can be retrieved to score the strategy quality.

Contrary to the approach proposed in [8] which measures strategy quality using the average investment degree over all simulated portfolios for each quarter t, we adopt a different perspective. We consider the Upper Confidence Bound (UCB) as a better indicator in order to prevent cash shortage situations due to cashflows variability. Indeed, driving the average investment degree close to the ideal $ID_t = 1$ for all t will automatically lead to situations in which some simulated portfolios will be overinvested ($ID_t > 1$) and thus to negative cash that should be avoided. Fig. 2 illustrates the proposed Upper Confidence Bound function to evaluate a strategy on a set of portfolios once simulation has finished. For each period t, the Upper Confidence Bound is defined as follows:

$$UCB(t) = E_p(ID_t) + 2\sigma_p(IDt_t) \qquad (4)$$

with $E_p(ID_t)$ representing the expected investment degree and $\sigma_p(IDt_t)$ the standard deviation at period t. The objective function obj measures the total deviation to the ideal $ID_t = 1.0$ and is defined as:

$$obj = \int_{t_1}^{t} |1.0 - UCB(t)| * d_t + K * (t_2 - t) \qquad (5)$$

We introduce a penalizing factor K for simulation ending prematurely. In the same vein, a portfolio with no cash to satisfy incoming contributions at $t <= t_2$ is terminated and gets thus penalized with the factor K. Note also that the integral in equation 5 is replaced by a discrete sum over all quarters of the active recommitment period, i.e., $t_1 \leq t \leq t_2$.
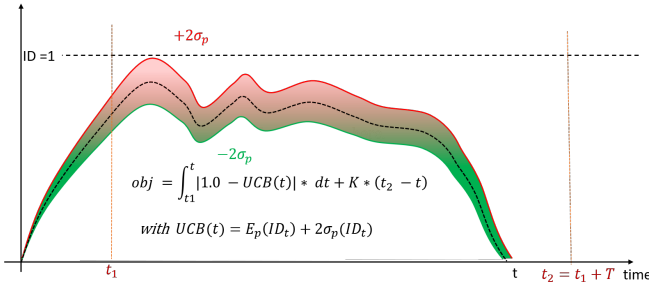
Fig. 2. Objective function using the Upper-Confidence Bound

C. Strategies evolution

Now that we have laid the foundation for scoring a strategy through portfolio's simulation, we are going to focus on their evolution. As aforementioned, a strategy is a mathematical function $f : \mathbb{R}^m \longrightarrow \mathbb{R}$ computing the amount of capital to recommit at quarter t. Our intent is to discover such a function using PE features at our disposal. These features can only reflect past and current evolution of the underlying funds. The resulting strategy, i.e., the function, will provide an approximation of the optimal amount of capital to recommit. In this work, it has been decided to search the best symbolic representations with regards to the objective function defined in the previous section. PE features described in Table I are thus represented as terminal symbols of an AST and connected by algebraic operators. Such a symbolic model is hence well suited for analysis by PE experts who can investigate its dynamic and increase results reliance.

TABLE I
FUNCTIONS AND TERMINAL SETS IMPLEMENTED IN THIS WORK

Name	Description
Operators	
+	Add two inputs
-	Subtract two inputs
*	Multiply two inputs
%	Divide two inputs with protection
min	Minimum b.t.w. two inputs
max	Maximum b.t.w. two inputs
Terminal sets/ Arguments	
C_t	Contributions at t
D_t	Distributions at t
ID_t	Investment degree at t
NAV_t	Net Asset Value at t
$error_t$	Deviation to ideal ID at t
$DZ^3(t)$	deZwart's strategy n°3 [8] at t
UC_{t-24}	Uncalled capital for commitments made 24 quarters ago
$CCommit_{t-24}$	Capital committed for 24 quarters

The search for improving strategies is performed according to a GP algorithm. The latter is described by Algorithm 1. At each generation, a population of recommitment strategies will be evaluated using simulations on a set of 250 initial portfolios, i.e., the training set. Evolutionary operators are applied on candidate strategies selected with regards to the objective function. A hall of fame records the best strategy over all generations. Once evolution reached the maximum number of generations ($NGEN$), the best encountered strategy during training is returned and then finally scored with a simulation

on a validation set of 1000 initial portfolios. Naturally, both training and validation set contain different portfolios. Only the results on the validation set are reported as it is done in classical machine learning scheme to show generalization capabilities of the resulting strategy.

Algorithm 1 Genetic programming hyper-heuristic

1: hall_of_fame ← create_empty_HOF(size=1)
2: population ← gen_ramped_half_and_half(NPOP,min,max)
3: **for** strategy in population **do**
4:     strategy.fitness ← simulate(strategy,training_instances)
5: **end for**
6: hall_of_fame ←update_HOF(hall_of_fame)
7: **while** gen ≤ NGEN **do**
8:     parents← selection(population)
9:     offsprings ← ∅
10:     **for** candidate_strategy in parents **do**
11:         **if** random() ≤ CXPB **then**
12:             mate ← sample(parents,1)
13:             offspring1,offspring2 ← crossover(candidate_strategy,mate)
14:             offsprings ← offsprings ∪ {offspring1,offspring2}
15:         **else if** random() ≤ CXPB+MUTPB **then**
16:             mutant ← mutation(candidate_strategy)
17:             offsprings ← offsprings ∪ {mutant}
18:         **else**
19:             repro_strategy ← copy(candidate_strategy)
20:             offsprings ← offsprings ∪ {repro_strategy}
21:         **end if**
22:     **end for**
23:     **for** new_strategy in offsprings **do**
24:         new_strategy.fitness ← simulate(new_strategy,training_instances)
25:     **end for**
26:     hall_of_fame ←update_HOF(hall_of_fame)
27:     population ← offsprings
28: **end while**
29: return hall_of_fame

V. EXPERIMENTAL SETUP

The experiments presented in this paper were carried out using the HPC facility of the University of Luxembourg [27]. The python library DEAP [12] has been considered for the GP implementation. A distributed implementation relying on a master-slave model has been put in place to evaluate each strategy of the population in parallel. Table II describes the GP hyperparameters while Table III described simulation parameters for both training and validation.

TABLE II
GP PARAMETERS

Runs	30
Generations	50
Population size	500
Crossover Probability (CXPB)	0.85
Mutation Probability (MUTPB)	0.1
Reproduction Probability	0.05
Tree initialization method	Ramped half-and-half
Selection Method	Tournament selection with size=7
Depth limitation	17
Crossover Operator	One crossover point
Mutation Operator	Grow

Note that simulation parameters are aligned with the one exposed in [8]. We consider quarterly recommitments and recommit every time to 4 funds selected proportionally to a fictive ESG (Environmental, Social Governance) score going beyond the scope of this paper and part of a wider study. This score has been computed with a specific correlation to the

Total Value Paid In (TVPI) which is an performance indicator for PE funds. Finally, we consider two different sets of initial portfolios. The first one consists of 250 portfolios for strategy evaluation during training and the second one is made up of 1000 portfolios for validating results and strategy performance.

Parameters	Training	Validation
Cashflows frequency	quarterly	quarterly
Investment period	26 years	26 years
Funds per recommitment	4	4
Fund selection	ESG score	ESG score
Number of simulated portfolios (per evaluation)	250	1000
Distributed simulation	True	False

VI. EXPERIMENTAL RESULTS

With respect to the parameters described in the previous section, 30 runs have been performed. Fig. 5 displays the average convergence curve obtained during the 50 generations. Convergence is very fast as it is generally observed in GP studies. The average function size, i.e., size of the strategy, first decreases rapidly too but steadily increase afterwards while the objective value reaches a plateau. We can clearly see that 15 generations would be enough and "bloating", i.e., phenomenon characterized by variable-length genomes gradually increasing in size during evolution impacts the scoring function training after this threshold. In fact, this bloating phenomenon is connected to the "overfitting" phenomenom in machine learning. It is good manner to stop the evolution to avoid the production of functions with no generalization capabilities. In the end, we aim at obtaining strategies which could be applied to any portfolios and cashflows.
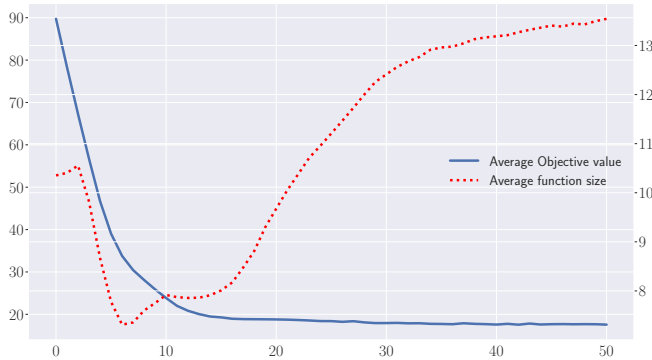

Fig. 3. Average convergence curve and average scoring function size over all 50 generations

For this reason, the best strategy obtained is tested on a validation set of 1000 different portfolios. Fig. 4 depicts the resulting investment degrees for this validation. The strategy $S^{best}(t)$ provided the best results during training. One can notice that this strategy is able to keep a high allocation during all the active investment period adjusting constantly recommitments. Some portfolios ($\approx 3.7\%$) became invalid during the simulations. These number would have been much larger if only the average investment degree had been taken into account. The Upper Confidence Bound relies on both average investment degree and standard deviation which permits to statistically bound the number of invalid portfolios to be at most 5%.
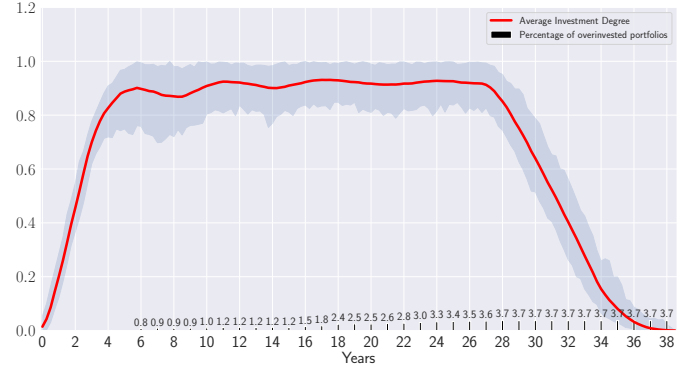

Fig. 4. Best strategy obtained from the 30 runs, i.e., $S^{best}(t) = \max\left(-Cash_t \times D_t + DZ^3(t), \min\left(Cash_t, D_t + 2UC_{t-24}\right)\right) + \min\left(Cash_t, \max\left(D_t^2, D_t + 2UC_{t24}\right)\right)$

One can also observe that the terminal/token $DZ^3(t)$ representing de Zwart's last strategy is part of $S^{best}(t)$. This empirically confirms the good results reported in [8] on a new dataset. Yet we can compare our results with the 3 strategies proposed by de Zwart et al. defined as follows:

- $DZ^1(t) = D_t$: recommit current distribution at t;
- $DZ^2(t) = D_t + UC_{t24}$: recommit distributions plus the uncalled capital made 24 quarters ago at t;
- $DZ^3(t) = \frac{1}{ID_t} \times (D_t + UC_{t24})$: Scale recommitment value obtained from $DZ^2(t)$ with the inverse of the investment degree.

As done for $S^{best}(t)$, the 3 strategies have been applied on the validation set made up of 1000 portfolios. Table VI reports the average investment degree, the standard deviation of the investment degree and the fraction of overinvested portfolios obtained for each strategy including S^{best}. With regards to the data used in this study, the 3 strategies $DZ^i(t)$ for $t \in \{1, 2, 3\}$ do not face any problem in terms of cash shortage. However, the highest investment degree reached by each of them is far from the ideal one leading to some "opportunity cost". Indeed $DZ^1(t)$ and $DZ^2(t)$ does not exceed 0.75 while $DZ^3(t)$ reached 0.8 with overcommitment. Besides, these values occur at the very beginning of the investment period and let us think that the initial overcommitment (30%) is the reason of these peaks. Fig. 5 represents graphically all 3 strategies including $S^{best}(t)$ confirms our first belief. We can clearly observe that the investment degree for $DZ^1(t)$ and $DZ^2(t)$ strongly decrease afterwards and stabilize later at nearly 0.6 while $DZ^3(t)$ gets stabilised at roughly 0.7.

On the contrary, $S^{best}(t)$ realizes 92% of the ideal investment degree. This value is reached after 15 years and is clearly not impacted by the initial overcommitment. Furthermore,

the investment degree stabilises very well around 0.9 and only starts decreasing when reaching the end of the active recommitment period. Fig. 5 illustrates perfectly the difference between these 4 strategies in terms of dynamics. Last but not least, the fraction of overinvested portfolios remains extremely low with $S^{best}(t)$. Due to the proximity with the ideal investment degree (ID=1) and cashflow's variability, it comes as no surprise that some portfolios get slightly overinvested. In this work, we notice that $S^{best}(t)$ only have 3.7% ($< 5\%$ threshold) of invalid portfolios at the end of the validation. Only LP investors can decide how to balance and evaluate their risk. Both opportunity cost and cash shortage are two conflicting objectives. Nonetheless, most investors diversify the investments and generally possess multi-class asset portfolios. If cash is missing, some liquid assets such as stocks and bonds could be sold to cope with such a situation. Of course, this solution should be tempered and only arise when no alternatives can be found. Please note that selling stocks and bonds still remain less critical than becoming a defaulting LP investor.
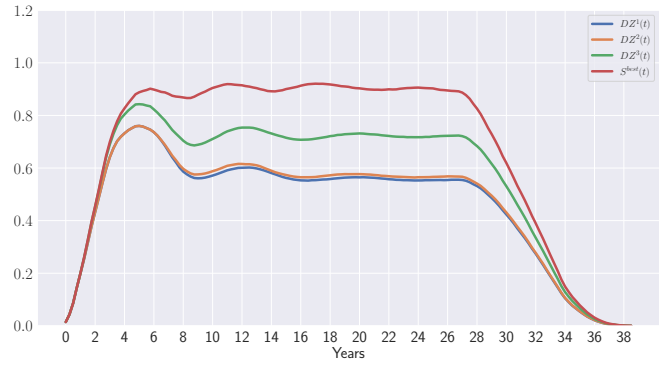

Fig. 5. Average investment degree for all 4 strategies

	$DZ^1(t)$			$DZ^2(t)$			$DZ^3(t)$			$S^{best}(t)$		
years	mean	std	proba	mean	std	proba	mean	std	proba	mean	std	proba
0	0.07	0.02	0.0	0.07	0.02	0.0	0.07	0.02	0.0	0.07	0.02	0.00
1	0.29	0.03	0.0	0.29	0.03	0.0	0.29	0.03	0.0	0.30	0.03	0.00
2	0.51	0.04	0.0	0.51	0.04	0.0	0.55	0.03	0.0	0.55	0.03	0.00
3	0.68	0.04	0.0	0.68	0.04	0.0	0.74	0.03	0.0	0.76	0.03	0.00
4	0.75	0.04	0.0	0.75	0.04	0.0	0.82	0.03	0.0	0.85	0.03	0.00
5	0.75	0.04	0.0	0.75	0.04	0.0	0.84	0.04	0.0	0.89	0.04	0.20
6	0.71	0.05	0.0	0.71	0.05	0.0	0.80	0.04	0.0	0.89	0.04	0.67
7	0.63	0.05	0.0	0.64	0.05	0.0	0.74	0.04	0.0	0.87	0.04	0.03
8	0.57	0.04	0.0	0.58	0.04	0.0	0.69	0.04	0.0	0.87	0.05	0.00
9	0.56	0.03	0.0	0.58	0.03	0.0	0.70	0.03	0.0	0.89	0.05	0.02
10	0.58	0.03	0.0	0.60	0.03	0.0	0.72	0.03	0.0	0.91	0.06	0.08
11	0.60	0.02	0.0	0.61	0.02	0.0	0.75	0.03	0.0	0.92	0.07	0.20
12	0.60	0.02	0.0	0.61	0.02	0.0	0.75	0.03	0.0	0.91	0.08	0.00
13	0.59	0.02	0.0	0.60	0.02	0.0	0.74	0.03	0.0	0.90	0.08	0.00
14	0.57	0.02	0.0	0.58	0.02	0.0	0.73	0.03	0.0	0.89	0.09	0.00
15	0.56	0.02	0.0	0.57	0.02	0.0	0.71	0.03	0.0	0.90	0.09	0.12
16	0.55	0.02	0.0	0.57	0.02	0.0	0.71	0.02	0.0	0.92	0.10	0.26
17	0.56	0.02	0.0	0.57	0.02	0.0	0.71	0.02	0.0	0.92	0.10	0.50
18	0.56	0.02	0.0	0.57	0.02	0.0	0.72	0.02	0.0	0.91	0.10	0.32
19	0.56	0.02	0.0	0.58	0.02	0.0	0.73	0.02	0.0	0.91	0.11	0.10
20	0.56	0.02	0.0	0.58	0.02	0.0	0.73	0.02	0.0	0.90	0.11	0.00
21	0.56	0.02	0.0	0.57	0.02	0.0	0.73	0.02	0.0	0.90	0.11	0.18
22	0.56	0.02	0.0	0.57	0.02	0.0	0.72	0.02	0.0	0.90	0.12	0.22
23	0.55	0.02	0.0	0.57	0.02	0.0	0.72	0.02	0.0	0.90	0.13	0.18
24	0.55	0.02	0.0	0.57	0.02	0.0	0.72	0.02	0.0	0.90	0.14	0.27
25	0.55	0.02	0.0	0.57	0.02	0.0	0.72	0.02	0.0	0.90	0.14	0.10
26	0.56	0.02	0.0	0.57	0.02	0.0	0.72	0.02	0.0	0.89	0.15	0.05
27	0.55	0.02	0.0	0.56	0.02	0.0	0.71	0.02	0.0	0.87	0.15	0.17
28	0.52	0.02	0.0	0.53	0.02	0.0	0.66	0.03	0.0	0.79	0.14	0.03
29	0.46	0.02	0.0	0.47	0.02	0.0	0.59	0.03	0.0	0.69	0.13	0.00
30	0.40	0.02	0.0	0.40	0.02	0.0	0.50	0.03	0.0	0.58	0.11	0.00
31	0.32	0.02	0.0	0.33	0.02	0.0	0.40	0.03	0.0	0.46	0.10	0.00
32	0.24	0.02	0.0	0.25	0.02	0.0	0.30	0.03	0.0	0.35	0.08	0.00
33	0.16	0.02	0.0	0.16	0.02	0.0	0.19	0.03	0.0	0.22	0.06	0.00
34	0.08	0.02	0.0	0.09	0.02	0.0	0.10	0.02	0.0	0.12	0.04	0.00
35	0.04	0.01	0.0	0.04	0.01	0.0	0.05	0.01	0.0	0.06	0.02	0.00
36	0.01	0.01	0.0	0.01	0.01	0.0	0.02	0.01	0.0	0.02	0.01	0.00
37	0.00	0.00	0.0	0.00	0.00	0.0	0.00	0.00	0.0	0.01	0.01	0.00
38	0.00	0.00	0.0	0.00	0.00	0.0	0.00	0.00	0.0	0.00	0.00	0.00

TABLE IV
SUMMARY STATISTICS OF THE INVESTMENT DEGREE IN RECOMMITMENT STRATEGIES

VII. CONCLUSION AND PERSPECTIVES

To maintain a high allocation to PE, LP investors have to carefully recommit to new funds. As of yet, arbitrary rules lacking robustness and flexibility are generally proposed as a current way out. Recently, some works have attempted to model and solve the PERP by proposing two main alternatives. The first one relies on cashflow predictions, i.e., statistically predicting future contributions and distributions while the second takes a more pragmatic approach building

approximation strategies only from past and current known features. In this work, we argue that these approximation strategies can be discovered using Evolutionary Learning, i.e., generating recommitment strategies with a GP algorithm. For this purpose, a simulation-based approach has been put in place to evaluate strategies over some investment period. Strategies are ranked according to the Upper Bound Confidence (UCB) to keep control on the fraction of portfolios with an investment degree higher than 1, i.e., invalid portfolios due to cash shortage. Results obtained after experiments show that the proposed approach leverages the generation of efficient recommitment strategies with higher investment degree while keeping within boundaries the number of invalid portfolios. Indeed, when getting closer to the ideal investment degree, the risk of defaulting slightly increases but remains below 5%. UCB allows LP investors to adjust the risk to be overinvested. Although the previous results confirm the suitability of automatising the generation of recommitment approaches, it could be more profitable to combine multiple strategies within a single simulation. This could take into account changes in market condition during portfolio's lifetime. This last remarks will be subject of next investigations. Finally, this work is a part of a larger project to develop an decision-aid tool for PE decision makers including ESG considerations. Indeed, the rise of Environmental, Social, and Governance (ESG) factors has been one of the major changes for PE partners over the last decade. It is therefore not trivial to evaluate them with a single criterion adding another level of difficulties for investors who needs to deal with fuzziness and conflicting objectives. Currently, there is no automatic and optimised solutions to help investors to maximise their allocation to ESG.

REFERENCES

[1] Thomas R Arnold, David C Ling, and Andy Naranjo. Waiting to be called: The impact of manager discretion and dry powder on private equity real estate returns. *The Journal of Portfolio Management*, 43(6):23–43, 2017.

[2] Anthony Brabazon, Michael O'Neill, and Seán McGarraghy. *Grammar-Based and Developmental Genetic Programming*, pages 345–356. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.

[3] Jurgen Branke, Su Nguyen, Christoph W. Pickardt, and Mengjie Zhang. Automated design of production scheduling heuristics: A review. *IEEE Transactions on Evolutionary Computation*, 20(1):110–124, 2016.

[4] Edmund K. Burke, Matthew R. Hyde, and Graham Kendall. Grammatical evolution of local search heuristics. *IEEE Transactions on Evolutionary Computation*, 16(3):406–417, 2012.

[5] J Heath Cardie, Katherine A Cattanach, and Mary Frances Kelly. How large should your commitment to private equity really be? *The Journal of Wealth Management*, 3(2):39–45, 2000.

[6] Boyuan Chen, Harvey Wu, Warren Mo, Ishanu Chattopadhyay, and Hod Lipson. Autostacker: A compositional evolutionary learning system. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '18, page 402–409, New York, NY, USA, 2018. Association for Computing Machinery.

[7] Etienne de Malherbe. Modeling private equity funds and private equity collateralised fund obligations. *International Journal of Theoretical and Applied Finance*, 07:193–230, 2004.

[8] Gerben de Zwart, Brian Frieser, and Dick van Dijk. Private equity recommitment strategies for institutional investors. *Financial Analysts Journal*, 68(3):81–99, 2012.

[9] Lynette R. D'mello, Aditya Jeswani, and J. Johnson. Stock price prediction using grammatical evolution. 2020.

[10] John H. Drake, Matthew Hyde, Ibrahim Khaled, and Ender Özcan. A genetic programming hyper-heuristic for the multidimensional knapsack problem. *Kybernetes*, 43(9/10):1500–1511, March 2014.

[11] G. Duflo, E. Kieffer, M. R. Brust, G. Danoy, and P. Bouvry. A gp hyper-heuristic approach for generating tsp heuristics. In *2019 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, pages 521–529, Los Alamitos, CA, USA, may 2019. IEEE Computer Society.

[12] Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau, and Christian Gagné. DEAP: Evolutionary algorithms made easy. *Journal of Machine Learning Research*, 13:2171–2175, jul 2012.

[13] Alex S. Fukunaga. Evolving local search heuristics for sat using genetic programming. In Kalyanmoy Deb, editor, *Genetic and Evolutionary Computation – GECCO 2004*, pages 483–494, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg.

[14] Emmanuel Kieffer, Grégoire Danoy, Matthias R. Brust, Pascal Bouvry, and Anass Nagih. Tackling large-scale and combinatorial bi-level problems with a genetic programming hyper-heuristic. *IEEE Transactions on Evolutionary Computation*, 24(1):44–56, 2020.

[23] Alejandro Sosa-Ascencio, Gabriela Ochoa, Hugo Terashima-Marin, and Santiago Enrique Conant-Pablos. Grammar-based generation of variable-selection heuristics for constraint satisfaction problems. *Genetic Programming and Evolvable Machines*, 17(2):119–144, 2016.

[15] E. Lara-Cárdenas, X. Sánchez-Díaz, I. Amaya, J. M. Cruz-Duarte, and J. C. Ortiz-Bayliss. A genetic programming framework for heuristic generation for the job-shop scheduling problem. In Lourdes Martínez-Villaseñor, Oscar Herrera-Alcántara, Hiram Ponce, and Félix A. Castro-Espinoza, editors, *Advances in Soft Computing*, pages 284–295, Cham, 2020. Springer International Publishing.

[16] Thomas Meyer. Hidden in plain sight—the impact of undrawn commitments. *The Journal of Alternative Investments*, 23(2):94–110, 2020.

[17] Daniel Nevins, Andrew Conner, and Greg McIntire. A portfolio management approach to determining private equity commitments. *The Journal of Alternative Investments*, 6(4):32–46, 2004.

[18] Adrian Oberli. Private equity asset allocation: How to recommit? *The Journal of Private Equity*, 18(2):9–22, 2015.

[19] Randal S. Olson and Jason H. Moore. *TPOT: A Tree-Based Pipeline Optimization Tool for Automating Machine Learning*, pages 151–160. Springer International Publishing, Cham, 2019.

[20] Mihai Oltean. Evolving evolutionary algorithms using linear genetic programming. *Evol. Comput.*, 13(3):387–410, 2005.

[21] Nasser R. Sabar, Masri Ayob, Graham Kendall, and Rong Qu. Grammatical evolution hyper-heuristic for combinatorial optimization problems. *IEEE Transactions on Evolutionary Computation*, 17(6):840–861, 2013.

[22] Nasser R. Sabar, Masri Ayob, Graham Kendall, and Rong Qu. Automatic design of a hyper-heuristic framework with gene expression programming for combinatorial optimization problems. *IEEE Transactions on Evolutionary Computation*, 19(3):309–325, 2015.

[24] Alejandro Sosa-Ascencio, Hugo Terashima-Marin, and Manuel Valenzuela-Rendon. Grammar-based genetic programming for evolving variable ordering heuristics. In *2013 IEEE Congress on Evolutionary Computation*. IEEE, 2013.

[25] Dean Takahashi and Seth Alexander. Illiquid alternative asset fund modeling. *The Journal of Portfolio Management*, 28(2):90–100, 2002.

[26] Rinde R. S. van Lon, Juergen Branke, and Tom Holvoet. Optimizing agents with genetic programming: an evaluation of hyper-heuristics in dynamic real-time logistics. *Genetic Programming and Evolvable Machines*, 19(1-2):93–120, 2017.

[27] S. Varrette, P. Bouvry, H. Cartiaux, and F. Georgatos. Management of an academic hpc cluster: The ul experience. In *Proc. of the 2014 Intl. Conf. on High Performance Computing & Simulation (HPCS 2014)*, pages 959–967, Bologna, Italy, July 2014. IEEE.

[28] Dennis G Wilson, Sylvain Cussat-Blanc, Hervé Luga, and Julian F Miller. Evolving simple programs for playing atari games. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '18, page 229–236, New York, NY, USA, 2018. Association for Computing Machinery.

[29] Jan Žegklitz and Petr Pošík. Benchmarking state-of-the-art symbolic regression algorithms. *Genetic Programming and Evolvable Machines*, 22:5–33, 03 2020.