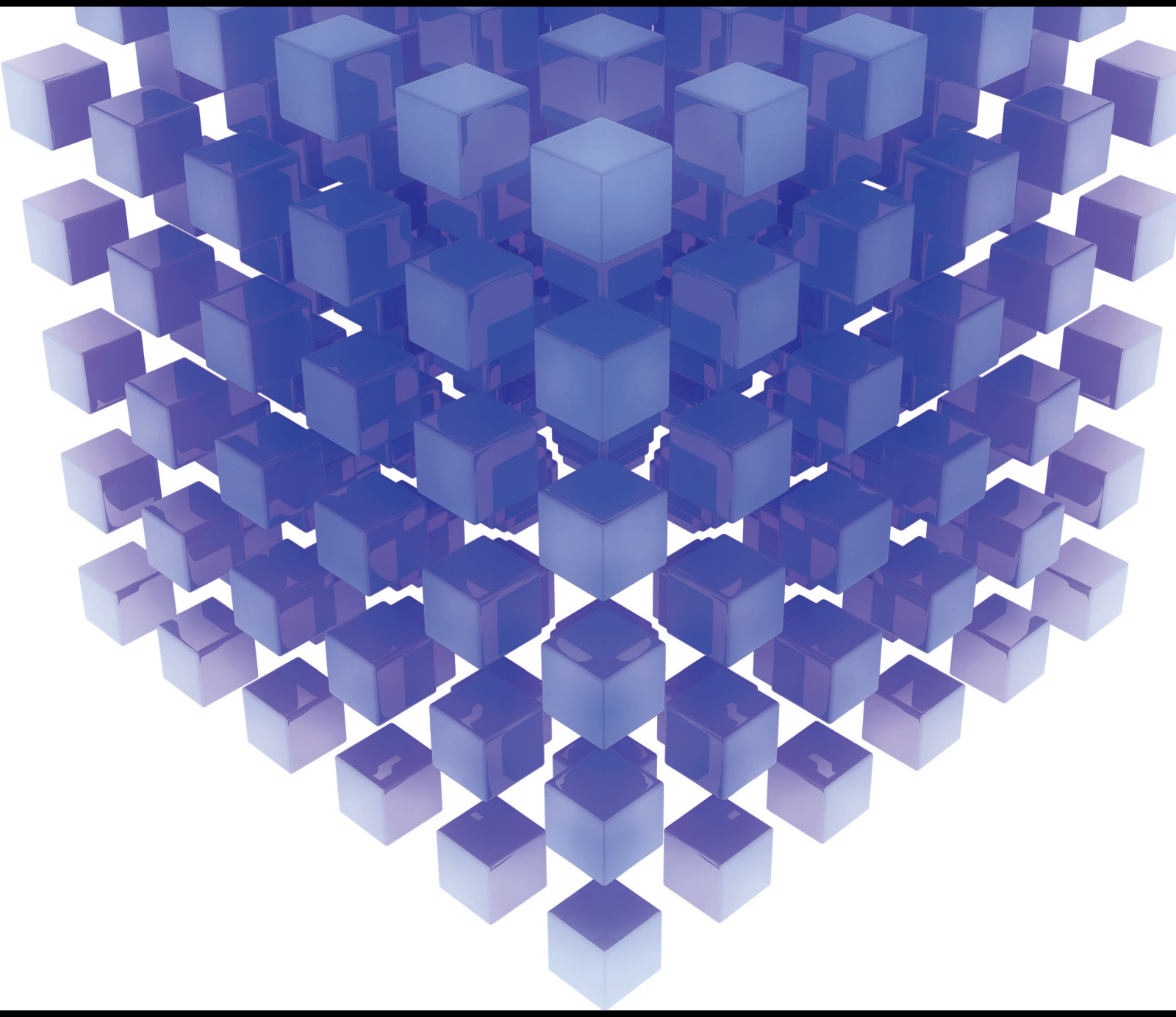
Mathematical Problems in Engineering

Optimization & Nature Inspired Computing

Lead Guest Editor: Srikanta Patnaik
Guest Editors: Hakim Bendjenna and Debahuti Mishra



Optimization & Nature Inspired Computing

Mathematical Problems in Engineering

Optimization & Nature Inspired Computing

Lead Guest Editor: Srikanta Patnaik
Guest Editors: Hakim Bendjenna and Debahuti Mishra

Contents

Hindawi
Mathematical Problems in Engineering
Volume 2024, Article ID 3427721, 21 pages
https://doi.org/10.1155/2024/3427721

Research Article

A Framework for Valuation and Portfolio Optimization of Venture Capital Deals with Contractual Terms

Mohammadreza Valaei and Vahid Khodakarami

Industrial Engineering Department, Bu-Ali Sina University, Hamedan, Iran

Correspondence should be addressed to Vahid Khodakarami; v.khodakarami@basu.ac.ir

Received 17 April 2023; Revised 20 November 2023; Accepted 20 December 2023; Published 25 January 2024

Academic Editor: Hakim Bendjenna

Venture capitalists invest not only in the business aspect of a deal but also in its contractual terms. Therefore, the selection of deals and the combination of contractual terms pose challenging decisions for them. This paper consists of two main sections. The first section introduces a novel framework for the valuation of venture capital (VC) deals, including startups and their contractual terms. By taking into account risk situations, this section presents the valuation of combined contractual terms, including call options, liquidity preference, and participant rights. In the second section, a new multiobjective mathematical model for VC deals and contractual terms portfolio selection is developed using right-tail probability, strategy alignment, and a utility function. To solve the proposed model, three metaheuristic algorithms—Non-Dominated Sorting Genetic Algorithm (NSGA-II), Multi-Objective Binary Harmony Search Algorithm, and Dynamic Tuning Parameter Binary Harmony Search Algorithm (DTPBHS)—are applied. Based on numerical examples, DTPBHS outperforms other algorithms in the "Mean Ideal Distance" index, but NSGA-II demonstrates the best performance in the "Rate of Achievement of two objectives simultaneously" index. Furthermore, we demonstrate that the proposed utility function is more robust than the right-tail probability function under default deals conditions.

1. Introduction

Venture capitalists, a specific type of financial intermediary, identify investees, such as startups, with significant potential for growth and entrepreneurial capacity. They provide these companies with funds, networking capabilities, and business skills to capitalize on market opportunities [1–3], playing a crucial role in the survival of new ventures [4].

On the one hand, investee selection involves not only evaluating venture projects but also considering contractual terms for venture capital (VCs). When selecting investees, VCs should assess factors such as the status of technology and the market, competitive position, growth strategy, and customer management [5]. This situation introduces a significant level of uncertainty and complexity in the valuation and selection of deals and their contractual terms. Therefore, VCs need to evaluate and maximize the profitability of investments by employing reliable and flexible methods for choosing investees and determining their contractual terms.

On the other hand, the investment strategies of VC firms play a primary role in the survival of new investments [6] and the selection of investees. These strategies may encompass the degree of VC participation in portfolio companies [7], as well as the level of diversification or localization.

In this paper, we will address three questions: How to value deals that combine contractual terms in uncertain conditions? How to model the portfolio selection of deals? What constitutes a robust objective function in the context of deal defaults?

Because the contractual terms of deals have a direct impact on investee selection, we propose a multidimensional method for the valuation of deals and their components, such as call options, liquidity preference, and participant rights. We apply the stochastic real options (ROs) method, stochastic DCF, and default rates to consider uncertainty and risk, offering an alternative to using risk-adjusted discount rates. Moreover, we introduce a new multiobjective mathematical model that incorporates both VC strategic and

financial objectives, considering uncertainty and constraints. The model helps in better-assessing risks and portfolio budgeting for deal selection, considering both financial and nonfinancial dimensions. Furthermore, we compare the performance of a utility selection approach to that of a right-tail selection approach when designing an index robust against deal defaults.

The main contributions of this paper include:

(i) Introducing a novel method for valuing deals and their contractual terms, incorporating default rates.
(ii) Developing a new multiobjective mathematical model for the portfolio selection of deals and contractual terms.
(iii) Employing a novel utility function for the selection of a VC portfolio and comparing its efficiency to the right-tail function in the context of a deal default scenario.

The remainder of this paper is organized as follows: In Section 2, we provide an overview of recent research on the valuation of deal contractual terms, VC strategy, and portfolio optimization. Section 3 explores the concepts of contractual terms valuation, VC portfolio selection, and strategic alignment. Additionally, we present a method for estimating the strategic misalignment of a deal with a VC strategy. In Section 4, we develop a meta-heuristic algorithm based on the harmony search (HS) algorithm to address the NP-hard computational complexity of the model for solving real-size problems. Section 5 discusses the applicability and results of the model using a numerical example, evaluating the performance of the proposed algorithms and the utility metric instead of the right-tail metric. Finally, in Section 6, we draw our conclusions and provide a summary to conclude the paper.

2. Literature Review

2.1. VC Contractual Terms Valuation. From the point of view of the information process perspective for VC decision-making, there are four stages: screening, origination, evaluation (due diligence), and negotiation [8]. Throughout these stages, qualitative and quantitative decision-making factors are hierarchically evaluated. Decision-making about valuation is one of the important challenges for VC investors because there are many uncertainties about the value of the deals. The valuation does not only include the startup's business of a deal but also contractual terms. Liquidity preference, investment amount, ownership stake, board control, option pool, prorata rights, valuation, antidilution, vesting, ownership stake, investment amount, participation, redemption rights, and dividends are negotiable and flexible contractual terms [9]. Some of these contractual terms have a direct effect on the valuation of a deal, such as the vesting situation, while others, like the option pool [10], have an indirect effect on the value of deals [11]. These contractual terms collectively influence the value of a deal. Therefore, a method is needed to evaluate both the business and contractual terms of deals under uncertain conditions.

There are several quantitative and qualitative methods for startup deal valuation, such as the Berkus method, Jordan Cooper, risk factor, VC methods, First Chicago, Chepakovich, multiples method, market-oriented approaches, score card, RO, and DCF [12–14]. Numerous of these methods are too qualitative and provide imprecise outputs (in some cases, unrealistic results) for startups.

RO valuation considers the cash flow value and adds the value of managerial flexibility to assets [15], as discussed by Li et al. [16], who proposed an RO valuation method for financial incentive allocation in infrastructural projects. They examined the effects of ambiguity on RO value based on pessimistic and optimistic approaches and concluded that the RO valuation method is better than the net present value (NPV) method for uncertainty analysis. As argued by Montajabiha et al. [17], the n-fold compound option model was used to evaluate multistages investment projects in R&D pharmaceutical projects. They also applied a robust model for R&D portfolio selection but did not utilize the value distribution function and the information on fat-tail distributions. Nigro et al. [18] assessed pharmaceutical industry projects by RO and categorized the project based on growth phases, and optimized the mathematical model, but they did not consider the uncertainty of valuations. One of the most popular methods for option pricing is the Black-Scholes model, which has been the subject of numerous discussions and examinations in academic literature [19]. There are different versions of the Black–Scholes model, such as literatures [20, 21], but the main difference in the proposed method compared to the literature is that the proposed valuation method is a combination of RO, DCF, decision tree, and Monte Carlo simulation [14].

2.2. VC Portfolio. Operation research problems usually answer only one of the "what," "where," and "when" questions [22]. This paper aims to address the combination of these questions for the VCs portfolio. Generally, VCs aim to select the best deals for success, but various risks can undermine profitability. As discussed in the review paper by Chaparro et al. [23], portfolio selection methods can be classified into 12 sectors: financial methods, probabilistic methods, option pricing theory, strategic methods, scoring methods, combinatorial optimization, behavioral methods, mapping approaches, ROs, integrated methods, information gap theory, and scenario-based approach. The paper suggests that the portfolio selection method should be based on the innovation level. When there is radical innovation, qualitative methods are preferred. As a result, the priority is behavioral and information gap theory. The second priority is integrated methods and ROs. In this paper, we consider that VCs do not invest in extreme radical innovation investees. Therefore, integrated methods and ROs are assumed to be appropriate.

According to Guo et al.'s research [24], different stakeholders have different tendencies toward portfolio selection goals, which are assumed in the proposed mathematical model. At the first level, top managers prefer to select a portfolio aligned with organizational strategies. At the second level, portfolio managers want to choose the projects

that maximize portfolio return. They pay great attention to the interrelation and synergies of projects. At the third level, the project managers should meet resource limitations and risk tolerance. In this paper, the proposed model belongs to the first and second levels of stakeholders. Aouni et al. [25] used the stochastic goal programing model and the concept of utility function in the case of 15 deals in order to design the decision-making process of VCs under three different scenarios. In another article published by Aouni et al. [26], a goal programing model based on fuzzy logic VC decision was proposed. The effect of the entry of a new deal on the existence of a portfolio and the effect of mutual interaction between new deals are considered by Zhao et al. [27]. Moreover, VCs can choose their deal portfolio based on the risk-taking or risk-aversion of investors by using a tuning parameter [27].

One of the most important criteria for VC portfolio optimization is portfolio size. According to Cumming [28], four categories of factors affect the size of a portfolio: (1) Characteristics of VC funds, including the type of fund (such as public or private VC), fund duration, fundraising, and the number of VC fund managers. (2) Characteristics of entrepreneurial companies, including development stage, technology, and geographical location. (3) The character of financing, including staging, syndication, and deal size. (4) Market conditions. In an area where the number of VCs is limited, entrepreneurs are likely not in a position to select among VC financiers [29]. Overall, the number of investees in a portfolio depends on many factors, such as funding stage, VC budget, VC type, the level of diversification, geographical distance, VC experience, financed entrepreneurial firms number, VC firm age, fundraising number and successful exits number [29, 30]. For example, Huntsman and Hoban [31] have demonstrated that ten investments may not be enough to reach a reasonably steady portfolio return for a VC portfolio, and portfolio diversity is not proportional to the fund size. Also, VC fund contractual conditions usually deny each investment from investing in more than a specific percentage (usually 10%) of the fund, and it is critical to maintain the appropriate diversity level [29]. In addition, the fund availability established by each investment company is seriously and directly associated with the fund size. As the investable companies grow, they are more likely to enter new funds. So, the small fund size restricts the VCs from accessing the advantages of investment results [29]. In the proposed model, the number and amount of investments are considered, and it is assumed that the stock offering of each deal is fixed for a certain amount of money because there are limited stock volume choices available for VCs to select. This assumption reflects the fact that investees, such as startup founders, prefer to keep a certain amount of equity to preserve their decision-making power in corporate governance.

Diversification criteria are another factor for portfolio selection. There are three principal categories of diversification (specialization): industry, stage of venture, and geography. Organizational learning theory discusses that specialized investor perseverance relies on extraordinary success in the field. However, diversified VCs are more likely to sustain due to less competition in a particular industry. Portfolio diversification is often beneficial in the early stages/high-risk stages. Informal venture capital (IVC) (including lesser-known individuals, entrepreneurial friends, and angel investors forming groups of IVCs) is less profited than formal venture capital (FVC) in terms of industry and stage diversity. Insufficient investment for IVC can push them to participate in fewer investment rounds, despite successful investment selections. It may prevent them from thoroughly enjoying the diversity benefits. The previous investment experience is suggested to provide a more robust alternative for IVC diversification needs. As a result, IVC can profit more by concentrating on the typical industry by achieving an expressive investment [32]. The small funds can protect forthcoming assets under management by adopting specialized strategies with relatively small investments in a restricted number of companies with losing diversification. In contrast, small funds may invest in many portfolio companies and choose a diversification strategy [29]. Therefore, in this article, according to the importance of this concept, the diversification index is used.

As discussed by Treville et al. [33], the worth of a VC portfolio is determined by the value of the small number of top deals that are available for assessment. Therefore, it can be acknowledged that the value of a portfolio relies on the presence of fat right-tail distribution deals rather than averages (IVC or FVC investor returns are extremely skewed, high probability of low-expected NPV and low probability of high-expected NPV, and existing literature confirms it [32]). The combination of the right tail of deals for the portfolio deal's probability distribution of NPV (we call it PDNPV) is different from the summation of deal value because this problem is not always subadditivity, no smooth, and nonconvex [34–36].

The "strategy" is the key to reaching profit for VCs [5]. As a result, the alignment of investee characteristics with the VC strategies is an important issue in selecting deals. Consequently, VC investors have strategies for reaching their goals. Ayob and Dana [37] proposed three strategies for VC products in developing countries, consisting of producing low-cost, differentiated, and specialized products. Park and Bae [38] proposed a 3D integrative framework of VC strategy in developing countries. The framework consists of three dimensions: target market, product/market maturity, and technological capability. They have also proposed seven practical strategies based on the main three strategies.

Various models exist for VC portfolio selection, including stochastic goal programing [25], fuzzy goal programing [26], modern portfolio theory [39], data envelopment analysis [40], and fuzzy inference system–agent [41]. Many research studies have focused on assessing the weight of selection criteria and integrating multicriteria decision-making techniques with fuzzy theory. However, only a limited number of studies have applied mathematical models specifically to the selection of VCs' investees (as opposed to deals). This limited usage underscores the untapped potential of mathematical models in the field of deal portfolio selection.

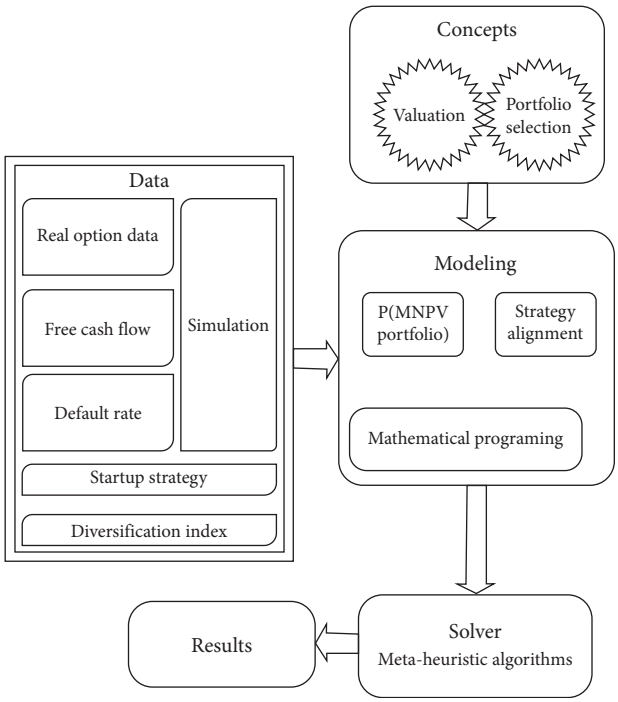


FIGURE 1: The framework of proposed venture capital portfolio selection.

To the best of our knowledge, no research has been conducted on the combined valuation and selection of startup deals and their contractual terms (such as call options, liquidity preference, and participant rights) for VC. Therefore, we propose a new method for valuation and a multi-objective portfolio selection mathematical model for VC, considering both financial and nonfinancial objectives. Various factors, including strategy alignment, the level of investee innovation, diversification, deal options, fund limitations, and the number of investments, are taken into account.

3. Problem Description and Formulation

The framework of this paper consists of two main concepts: valuation and selection of VC portfolio (Figure 1).

In Figure 1, we apply valuation and portfolio selection concepts to design a new model for deal contractual terms valuation and portfolio optimization. For valuation, the input data involve free cash flows (FCFs) probability function, default rates in different periods, the data related to ROs, and contractual terms. The inserted data for portfolio selection includes VC strategies, diversification index, the minimum NPV required for VC, valuation of deals and their contractual terms, budget constraints, and the maximum number of investees. Then, a solver method like a Meta-heuristic algorithm will determine the portfolio's compound.

3.1. The Valuation of VC Contractual Terms. In this paper, it is assumed that there are several deals with a variety of contractual terms. Therefore, we not only evaluate the NPV of a deal but also assess the effect of deals with contractual terms between the founder and the investors. We consider all kinds of deal risks into three groups: hard factors, soft factors, and scenario factors.

The factors with sufficient data about future scenarios and probabilities are classified as hard factors. In this group, these factors will be estimated using decision trees and probability distribution functions. The second group consists of scenario factors, where the scenarios are identifiable, but their probabilities cannot be determined directly. Here, we employ the RO method to account for them. In the third group, soft factors are typically examined qualitatively, including individual and managerial attributes of the investor, the motivation and experience of the founders, the team members' ability to recover in case of partial failures, and their resilience in the face of economic, political, social, and legal conditions [42]. The evaluation of these soft factors is carried out using expert-elicited Bayesian network (EBN) analysis, as described by Valaei and Khodakarami [42]. The output of EBNs is the default rate in each period of a deal's life.

In the following, By using an example, we will present how to calculate the value of a stochastic European call option by using a Monte Carlo Simulation for a startup deal. Then, the formula for the valuation of contractual terms, specifically for liquidity preference and participant rights, is proposed.

In this paper, the price of a European call option is calculated based on the Black–Scholes model (Equation (1)) (it is evident that various alternative methods exist for RO pricing that can be used, including Binomial Models) [43]. The inputs of the model consist of the current underlying asset price, the strike underlying asset price, the time to maturity, the risk-free rate, and the volatility of the underlying asset [43].

$$O_t = S \cdot N(d_1) - Xe^{-\alpha t} \cdot N(d_2); \quad d_1 = \frac{\ln\left(\frac{S}{X}\right) + \left(\alpha + \frac{\sigma^2}{2}\right)t}{\sigma\sqrt{t}}; \quad d_2 = d_1 - \sigma\sqrt{t}, \tag{1}$$

The parameters of the valuation model are described as follows:

(O_t) Call option premium at time t*—the value of a call option with expiration time "t";

(V_t) Future FCF value*—future FCF value of deal at time "t";

(R_t) Realized FCF value *—the FCF value of a deal at time "t";

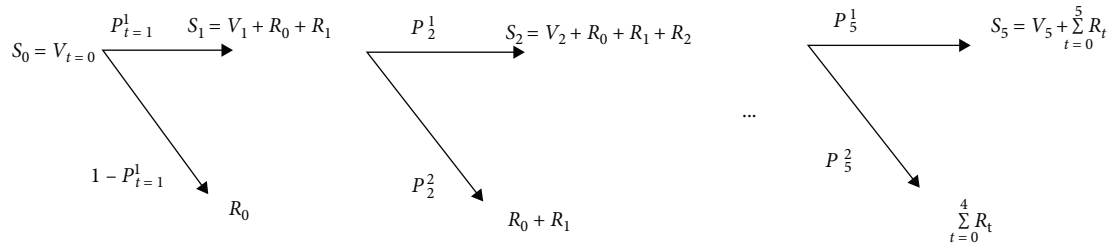


Figure 2: A simple NPV decision tree of a deal.

(S_t) FCF value of the deal at time "t" *—overall past and future value of FCF at time "t" related to stock ownership. It sums of V_t and $\sum R_t$;

(X_t) Strike price of deal *—the exercise price is the value at which the option can be exercised at time "t";

(g) Time to maturity of call option;

(α) Risk-free rate;

(σ) Volatility—the underlying asset volatility;

$N()$ Cumulative normal density function N (0.1);

(O_g) Call option premium *—the value of a call option with expiration time g;

($P^2{}_t$) Failure probability—default rate at time "t";

($P^1{}_t$) No failure probability at time "t" ($P^1{}_t = 1 - P^2{}_t$);

(w) Share ownership percentage;

(L_g) liquidity preference multiple;

*If there is a sign of "′" such as O'_t, it shows the present value of each parameter.

For instance, there is a probability distribution of net present value (PDNPV) for a deal. Because the NPV of a deal is not a single value, the NPV of derivative tools, such as a call option, will not be a singular value either. Therefore, in this paper, we employ the Monte Carlo simulation method to generate PDNPV for deals and their options, reflecting the premium cost in the cash flow of a deal.

In this paper, the value of the current underlying asset is a probability distribution function. Assuming that the NPV motion follows a random walk based on Geometric Brownian Motion, as described in Equation (2) [44], We calculate the standard NPV of a new deal (without considering the default rate) with a call option using Equation (3).

$$S_t = S_0 \exp\left[\left(\mu - \frac{\sigma^2}{2}\right)\Delta t + \sigma\sqrt{\Delta t}\varepsilon\right], \tag{2}$$

$$\text{Deal NPV}_t = \begin{cases} -O_t & S_t - X_t < 0 \\ S'_5 - X'_t - O_t & S_t - X_t \geq 0 \end{cases}. \tag{3}$$

Based on Equation (3), if the price of the deal is higher than the strike price, the call option will be exercised. Otherwise, the investment will not continue, and the deal will be abandoned, resulting in the NPV of the deal will be negative, equal to the option premium. For instance, Figure 2 shows the PDNPV of a deal with a European call option and a time to maturity of 4 years, demonstrated using a Monte Carlo simulation (The Monte Carlo simulation is done by Crystal Ball software). If there is a failure probability in each period ($1 - P_t$), the NPV of the startup is shown in Figure 2.

The NPV of a startup deal with a call option (with expiration g year) and the default rate is shown in Equation (4).

$$\text{Deal NPV} = \begin{cases} -O_g & \left(\prod_{t=1}^{T=g} P^1_t\right)\left(V_g + \sum_{t=1}^{g} R_t\right) - X_g < 0 \\ \left(\prod_{t=1}^{T=5} P^1_t\right)\left(V'_{T=5} + \sum_{t=1}^{T=5} R'_t\right) + \sum_{\tau=g+1}^{T}\left(\left(\prod_{t=2}^{\tau} P^1_{t-1}\right)(1 - P^1_\tau)\left(\sum_{t=1}^{\tau-1} R'_t\right)\right) - X'_g - O_g & \left(\prod_{t=1}^{T=g} P^1_t\right)\left(V_g + \sum_{t=1}^{g} R_t\right) - X_g \geq 0 \end{cases}, \tag{4}$$

The NPV of a startup with a call option (with expiration g), default rate, and liquidity preference is shown in Equation (5).

$$\text{Deal NPV}=\begin{cases} -O_g & \left(\prod_{t=1}^{T=g} P_t^1\right)\left(V_g+\sum_{t=1}^{g} R_t\right)-X_g<0 \\ \max\left\{\begin{array}{c}\left(\prod_{t=1}^{T=5} P_t^1\right)\left(V'_{T=5}+\sum_{t=1}^{T=5} R'_t\right), \\ \left(\prod_{t=1}^{T=5} P_t^1\right)\cdot\min\left\{\left(L_g\times X'_g\right),\left(\dfrac{\left(V'_{T=5}+\sum_{t=1}^{T=5} R'_t\right)}{w}\right)\right\}\end{array}\right\}+ & \left(\prod_{t=1}^{T=g} P_t^1\right)\left(V_g+\sum_{t=1}^{g} R_t\right)-X_g\geq 0,\quad g<5 \\ \sum_{\tau=g+1}^{T=5}\left(\left(\prod_{t=2}^{\tau} P_{t-1}^1\right)(1-P_\tau^1)\cdot\min\left\{\left(L_g\times X'_g\right),\left(\dfrac{\sum_{t=1}^{\tau-1} R'_t}{w}\right)\right\}\right)-X'_g-O_g & \end{cases}. \tag{5}$$

The NPV of a startup with a call option (with expiration g), default rate, liquidity preference, and participant right is shown in Equation (6). To assess the impact of certain independent contractual terms, please refer to reference [11].

$$\text{Deal NPV}=\begin{cases} -O_g & \left(\prod_{t=1}^{T=g} P_t^1\right)\left(V_g+\sum_{t=1}^{g} R_t\right)-X_g<0 \\ \left(\left(\prod_{t=1}^{T=5} P_t^1\right)\cdot\left(\min\left\{\left(L_g\times X'_g\right),\left(\dfrac{V'_5+\sum_{t=1}^{T=5} R'_t}{w}\right)\right\}+w\times\max\left(0,\dfrac{V'_5+\sum_{t=1}^{T=5} R'_t}{w}-\left(L_g\times X'_g\right)\right)\right)\right)+ & \left(\prod_{t=1}^{T=g} P_t^1\right)\left(V_g+\sum_{t=1}^{g} R_t\right)-X_g\geq 0,\quad g<5 \\ \sum_{\tau=g+1}^{T=5}\left(\left(\prod_{t=2}^{\tau} P_{t-1}^1\right)(1-P_\tau^1)\left(\min\left\{\left(L_g\times X'_g\right),\left(\dfrac{\sum_{t=1}^{\tau-1} R'_t}{w}\right)\right\}+w\times\max\left\{0,\left(\dfrac{\sum_{t=1}^{\tau-1} R'_t}{w}\right)-\left(L_g\times X'_g\right)\right\}\right)\right)-X'_g-O_g & \end{cases}. \tag{6}$$

3.2. VC Portfolio Selection. The mathematical model in this paper is based on the multidimensional multiple-choice knapsack problem (MMKP) [45] and the portfolio selection problem under the value-at-risk measure [36, 46], which does not have the subadditivity property. MMKP is a generalization of the ordinary knapsack problem [47], and it is a more complex variant of the binary knapsack problem and computationally is an NP-Hard problem [45, 48, 49]. Moreover, the NP-Hard complexity of portfolio selection problem with Value at Risk objective (such as right tail probability) is proved by Benati and Rizzi [46]. Since the real-world investment problems are composed of different objectives, it is proposed a multiobjective stochastic version of the multidimensional multichoice knapsack problem. Thus, the proposed model is a multiobjective binary model to select a set of deals and their RO, maximizing the right tail of a cumulative probability of the portfolio NPV and minimizing the strategic misalignment of the selected deals.

In this section, the proposed mathematical model is introduced. The following assumptions are made:

(i) The cash flow predictions of each deal are independent of each other.
(ii) It is assumed that the stock offering of deals is fixed for a certain amount of fixed money.
(iii) The portfolio's budget is deterministic, and it is divided into consuming and reserving budgets.
(iv) The focus of this paper is limited to a specific geographical area (without geographical diversification).
(v) The FCFs are predicted stochastically.

In the following, decision variables are determined, and the abbreviations and parameters used in the mathematical model are defined.

3.2.1. Model Indices. Investee: i;

Industry: j;

Deals: u (all kinds of contractual terms for each deal are considered independent variables).

3.2.2. Model Parameters. Z_1: The first objective function is designed for calculating the right tail of the portfolio NPV distribution;

Z_2: The second objective function aims to quantify the strategy misalignment;

MNPV: The minimum attractive NPV for the VC portfolio;

D_{ji}: The total amount of strategy misalignment for investee i in industry j;

d_{ji}^{y}: The amount of strategy misalignment for investee i in industry j regarding strategic dimension y;

U: The maximum number of company strategy dimensions;

C_{jiu}: The investment cost for the investee i in industry j with deal u;

P_{jiu}: The present value of future cost for the investee i in industry j with deal u;

B: The maximum available budget;

E: The maximum reserved available budget;

N: The maximum number of the deals that VC can manage;

L: The minimum amount of Herfindahl diversification index for VC portfolio;

H: The maximum amount of Herfindahl diversification index for VC portfolio;

k: The maximum number of industries;

n_j: The maximum number of investee in industry j;

T_{ji}: The maximum number of deals for the investee i in industry j;

NPV_{jiu}: NPV of the investee i in industry j with deal u;

$\Phi_{jiu}(\mathrm{MNPV})$: The complementary cumulative distribution function of the NPV of investee i in industry j with deal u for the MNPV;

$f^{\sum_{j=1}^{k}\sum_{i=1}^{n_j}\sum_{u=1}^{T_{ji}}\chi_{jiu}}(h)$: $f(h)$ is a joint probability distribution function for the sum of the probability distribution function of $\sum_{j=1}^{k}\sum_{i=1}^{n_j}\sum_{u=1}^{T_{ji}}\chi_{jiu}$.

3.2.3. Decision Variables. χ_{jiu}: Binary variable for selecting the deal i in industry j;

α_{jiu}: Auxiliary binary variable for selecting the deal i in industry j.

To evaluate the strategic misalignment between investees and VCs, a distinct version of the index, influenced by literatures [50, 51] and incorporating the Herfindahl index [52, 53], is employed. This index measures the Euclidean Distance for each dimension. The formulations are as follows (Equation (7)):

$$D_{ji} = \sqrt{\sum_{y=1}^{U=9} \frac{\left(d_{ji}^{y}\right)^2}{U-1}},\ d_{ji}^{y} = \frac{\text{Investee i strategy index in dimantion of } y - \text{ VC strategy index in dimantion of } y}{\text{Maximun strategy index in dimantion of } y - \text{Minimum strategy index in dimantion of } y}, \tag{7}$$

where D_{ji} represents the overall strategic misalignment of deal i in industry j, and d_{ji}^{y} is the strategic misalignment of deal i in industry j at dimension y.

As given in Table 1, For instance, if a VC with an "import substitution" strategy intends to invest in a deal with an "early market entry" strategy, conflicts may arise in strategic alignment [38].

In order to show the formulas, in Equations (8)–(17), we illustrate the complete mathematical model as follows:

Max Z_1:

$$\sum_{j=1}^{k}\sum_{i=1}^{n_j}\sum_{t=1}^{T_{ji}} \Phi_{jiu}(\mathrm{MNPV})\alpha_{jiu} + \sum_{j=1}^{k}\sum_{i=1}^{n_j}\sum_{t=1}^{T_{ji}} \int_{-\infty}^{\mathrm{MNPV}} \left(\Phi_{jiu}(\mathrm{MNPV}-h)\cdot\chi_{jiu}\right) f^{\left(\chi_{1,1,1},\ldots,\chi_{k,n_j,T_{ji}}\right)}(h)\, dh, \tag{8}$$

$$\text{Min } Z_2: \frac{\sum_{j=1}^{k}\sum_{i=1}^{n_j} D_{ji}\left(\sum_{u=1}^{T_{ji}}\chi_{jiu}\right)}{\sum_{j=1}^{k}\sum_{i=1}^{n_j}\sum_{u=1}^{T_{ji}}\chi_{jiu}}, \tag{9}$$

$$\sum_{j=1}^{k}\sum_{i=1}^{n_j}\sum_{t=1}^{T_{ji}} C_{jiu}\left(\chi_{jiu}+\alpha_{jiu}\right) \le B, \tag{10}$$

$$\sum_{j=1}^{k}\sum_{i=1}^{n_j}\sum_{t=1}^{T_{ji}} P_{jiu}\left(\chi_{jiu}+\alpha_{jiu}\right) \le E, \tag{11}$$

$$1 \le \sum_{j=1}^{k}\sum_{i=1}^{n_j}\sum_{u=1}^{T_{ji}} \left(\chi_{jiu}+\alpha_{jiu}\right) \le N, \tag{12}$$

Table 1: Strategy alignment measurement [38].

	Strategies							Strategic Misalignment Measurement
	Reactive imitation	Proactive localization	Import substitution*	Creative imitation	Early market entry*	Global niche	Global innovation	d_{ji}^{y*}
Dimension								
Product-market								
Maturity	Existing:1	Emerging:0	**Existing:1**	Emerging (local/exsting (global):0	**Emerging:0**	Existing:1	Emerging:0	$0-1=-1$
Growth potential	Low:1	Moderate:2	**Low:1**	Moderate/high:2 /3	**Very high:4**	High:3	Very High:4	$\frac{4-1}{4-1}=1$
Market uncertainty	Low:1	Moderate:2	**Low:1**	Moderate:2	**High:3**	Moderate:2	Very High: 4	$\frac{3-1}{4-1}=0.66$
Technological capability								
Technological capability, Tech. Capabilities (pioneer/follower)	Local follower:0	Local follower:0	**Local pioneer:1**	Local pioneer/global follower:0	**Global follower:0**	Global pioneer:0	Global pioneer:0	$0-1=-1$
R&D investment	Low:1	Low:1	**Moderate:2**	Moderate:2	**Moderate:2**	High:3	High:3	$\frac{2-2}{3-1}=0$
Technological partners	–	Global:1	**Local:0**	Local/ global:0 or 1	**Global:1**	Global:1	Global:1	$1-0=1$
Priority of R&D	–	–	**Cost reduction:1**	Product quality/cost reduction:0 or 1	**Entry timing:0**	Product quality	Entry timing	$0-1=-1$
Target market								
Target market	Local:0	Local:0	**Local:0**	Local/global:0 or 1	**Global:1**	Global:1	Global:1	$1-0=1$
Degree of internationalization	Low:1	Low:1	**Low:1**	Moderate:2	**High:3**	High:3	High:3	$\frac{3-1}{3-1}=1$

*For example, if the assumed strategy of a VC is Import Substitution and the strategy of an investee is Early Market Entry, the misalignment strategy for the selection is calculated in the last column. Bold columns are examples of strategies and are relevant.

Table 2: Coding of solutions.

Industry 1								Industry 2								Industry n							
Investee 1				Investee 2				Investee 1				Investee 2				Investee 1				Investee 2			
Deal 1	Deal 2	Deal 3	Deal 4	Deal 1	Deal 2	Deal 3	Deal 4	Deal 1	Deal 2	Deal 3	Deal 4	Deal 1	Deal 2	Deal 3	Deal 4	Deal 1	Deal 2	Deal 3	Deal 4	Deal 1	Deal 2	Deal 3	Deal 4
0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

$$L \leq 1 - \left(\frac{\left(\sum_{j=1}^{k} \left(\sum_{i=1}^{n_j} \sum_{u=1}^{T_{ji}} \left(\chi_{jiu} + \alpha_{jiu} \right) \right) \right)^2}{\left(\sum_{j=1}^{k} \sum_{i=1}^{n_j} \sum_{u=1}^{T_{ji}} \left(\chi_{jiu} + \alpha_{jiu} \right) \right)^2} \right) \leq H, \tag{13}$$

$$\sum_{u=1}^{T_{ji}} \left(\chi_{jiu} + \alpha_{jiu} \right) \leq 1 j = 1, \ldots, k \quad i = 1, \ldots, n_j, \tag{14}$$

$$\chi_{jiu} + \alpha_{jiu} \leq 1 j = 1, \ldots, k\,, \quad i = 1, \ldots, n_j\,, \quad u = 1, \ldots T_{ji}, \tag{15}$$

$$\sum_{j=1}^{k} \sum_{i=1}^{n_j} \sum_{u=1}^{T_{ji}} \alpha_{jiu} = 1, \tag{16}$$

$$\chi_{jiu} = \{0, 1\}. \tag{17}$$

As we aim to capitalize on the asymmetric returns of deals, the first objective (8) employs the right tail of PDNPV at the point of MNPV. This aligns with the opportunistic strategy of seeking the highest investment return. Proceeding from the left side of the distribution, we utilize numerical integration with a trapezoidal base and calculate the cumulative probabilistic distribution functions until reaching the value of MNPV (its proof is presented in the Appendix [54]). The second objective (9) focuses on estimating the strategic misalignment of a portfolio, resulting in a value of one or greater.

Constraints (10) and (11) are related to budget considerations. Constraint (10) pertains to the maximum budget available for selecting deals right now, while Constraint (11) entails reserving a budget for the future cost of the selected deal, including the exercise of options in subsequent years.

Additionally, Constraint (12) establishes boundaries for both the maximum and minimum number of investees that can be managed, resembling a cardinality constraint [55]. In this research, industrial diversification is considered due to the importance of the industrial type of investees that operate. A diversification index called the Herfindahl index [52, 53] is applied in Constraint (13) on the portfolio selection model. It is proposed that the upper and lower bounds of the constraint be determined based on successful and similar VC practices (see [56]).

Since VCs can choose only one deal for each investee, we use Constraint (14). Constraint (15) is intended to decide between the primary and auxiliary variable choices. In the first part of the primary objective function, you must select one of the deals. Therefore, we add Constraint (16) to the model. Constraint (17) represents binary variables. Each variable shows the value of a deal with individual contractual terms (consisting of options).

4. Algorithm Design

The number of variables in the proposed problem could increase significantly when a VC has several investees with different contractual terms. The exact algorithms, such as branch and bond algorithms and dynamic programing algorithms, cannot handle large-scale problems, and solving them requires a significant amount of CPU time due to their NP-hardness [57]. Given these challenging, several meta-heuristic algorithms have been developed for optimization in the last few decades [58]. As discussed by Zou et al. [59], the HS algorithm is a well-organized method for solving knapsack problems, and it can find better solutions compared to other metaheuristics, such as genetic algorithm in a stochastic multiobjective problem under the same situation [60, 61]. The HS method is an emerging metaheuristic optimization algorithm employed to cope with numerous challenging tasks during the past decade. The HS algorithm is inspired by the underlying principles of musicians' improvization of harmony, which can be seen as a search process for the global optimum in optimization problems that are evaluated by an objective function [62]. The musical harmony in HS is similar to a variable vector, and the best harmony achieved in the end can be seen as the optimal global solution [58].

4.1. Dynamic Parameters Tuning Multiobjective HS. A variety of harmony algorithms have been proposed, including an efficient HS algorithm [63], an improved HS algorithm [64], multiobjective binary differential evolution HS (MOBDEHS) [65], and multiobjective HS algorithm [66]. A new version of the multiobjective binary HS (MBHS) algorithm [66] is the dynamic parameters tuning multiobjective harmony search algorithm (DPTMOHS), which uses dynamically tuned parameters to create nondominant solutions. The algorithm for dynamically tuning the parameters has been designed based on reference [60], and the Pseudocode of the proposed algorithm (DPTMOHS) is shown in the coding done by MATLAB 2013a software on a Core i7, 8 GB. Before delving into the detailed explanation of the algorithm, it is necessary to present the coding method of the problem, as shown in Table 2.

As depicted in Figure 3, a deal is selected from the harmony memory (based on the elitism parameter) if a new random number is less than the harmony memory consideration rate (HMCR). Subsequently, at the next level, if a new random number is less than the pitch adjustment rate (PAR) (the mutation parameter), a deal is randomly selected from the neighborhood of the last deal position (with all deals sorted in ascending order). If the algorithm does not enter the process of HMCR and PAR, it utilizes a random mechanism, as shown in Figure 3.

```
For each X_i ∈ [ i = 1, ..., n ] do
    If rand () =< HMCR then
            Select from memory of harmony
        If rand () =< PAR then
            Select the neighbouring deal of your choice.
        End
    Else
        If rand ()>0.5
            Randomly select a deal
        Else
            Do not select any deal
        End
    End
End
```

FIGURE 3: Pseudocode of algorithm.

To handle the constraint mentioned in the model within the algorithm, a pruning approach is employed for newly generated solutions. In essence, if an infeasible solution is generated, the algorithm promptly discards it.

4.2. Comparing Metric. There are various metrics for comparing approximate algorithms when accuracy is unmeasurable. In this research, we employ three metrics to compare the three algorithms. These metrics include the rate of achievement of two objectives simultaneously. The relative achieved spread (RAS), which measures the rate of achieving the two best objectives simultaneously, the mean ideal distance (MID, which quantifies the distance between Pareto solutions and the best answer), and the spread of nondominated solutions (SDS, indicating the range of nonexhaustive solutions) [67]. In Equations (18)–(20), Z_{1s} and Z_{2s} are normalized using the method described in literature [67], and n shows the number of solutions in Pareto solutions.

$$\mathrm{MID} = \frac{\sum_{s=1}^{n} C_s}{n} C_s = \sqrt{(Z_{1s} - Z_1^*)^2 \quad + \quad (Z_{2s} - Z_2^*)^2}, \quad (18)$$

$$\mathrm{SDS} = \sqrt{\frac{\sum_{s=1}^{n} (\mathrm{MID} - C_s)^2}{n-1}}, \quad (19)$$

$$\mathrm{RAS} = \frac{\sum_{s=1}^{n} \left(\frac{Z_{1s} - \min\{Z_{1s}, Z_{2s}\}}{\min\{Z_{1s}, Z_{2s}\}} \right) + \left(\frac{Z_{2s} - \min\{Z_{1s}, Z_{2s}\}}{\min\{Z_{1s}, Z_{2s}\}} \right)}{n}. \quad (20)$$

5. Computational Tests

5.1. Generating Test Problems. To test and verify the algorithms in the proposed model, we generate practical testing by using 10 deals with 40 premium call options (Table 3). The strategy misalignment of each candidate deal is randomly generated from a uniform distribution (1, 5), as demonstrated in Table 4. The cumulative probability distributions of NPV are summarized in Table 5.

5.2. Comparing Results. To assess the performance of the DPTMOHS algorithm, we utilize two series of test problems: one for small size and another for medium size.

5.2.1. Small-Size Test Problem. For the small-sized problem, new test problems were designed to evaluate the capability of the DPTMOHS algorithm in finding Pareto solutions. Three groups of problems, with sizes 8, 10, and 12 and each having three levels of MNPV (with values of 50, 75, and 100), were created. The results, based on quality metrics (the proportion of Pareto solutions [68]), are presented for 1,000 iterations of the algorithm in Table 6. The table illustrates that the DPTMOHS algorithm can effectively find Pareto solutions, demonstrating the efficacy of the proposed algorithm in identifying optimal points for the problem (the data are provided in Table 7).

5.2.2. Medium-Size Test Problem. The algorithm of Nondominated Sorting Genetic Algorithm II (NSGA-II) [69, 70] and MBHS algorithm [66] are used for comparison with the DPTMOHS algorithm. These algorithms need to adjust their parameters (DPTMOHS does not need to parameter tuning). In this paper, the NSGA-II (the maximum number of iteration parameters is not considered because the termination condition is determined based on the number of objective evaluations) and MBHS (the harmony memory size (HMS) parameter in the classic harmony search algorithm has been omitted in this study. This omission is based on the assumption that what is crucial for VCs is to generate a range of dominant solutions rather than focusing on a single solution. Furthermore, due to the utilization of dynamic parameter tuning, the count of nondominated solutions may fluctuate throughout the optimization process) parameters [66] are tuned by the Taguchi method for tuning parameters [60]. The tables are selected for standard orthogonal array L_{16} (4^2) and L_9 (3^3) experiments, in which three repetitions are performed (the data are provided in Tables 8 and 9). The levels of experiments were conducted using the Taguchi method, and their results were obtained based on the signal-to-noise (S/N) ratio. Then, the best-tuned parameters of the algorithm were calculated using Minitab 17 software [60]. Tables 10 and 11 show the best parameters of the MBHS and NSGA-II algorithms.

After determining the parameters for both the MBHS and NSGA-II algorithms, their performance will be assessed against that of the DPTMOHS algorithm, utilizing RAS, MID, and SDS metrics (see Table 12).

To determine whether there exists a statistically significant difference in the performance of the algorithms (Table 12), we employ the Kruskal–Wallis test (Figure 4) [67] and a paired t-test (Table 13) [68] with a confidence interval of 0.95.

The results of the Kruskal–Wallis test indicate no significant difference in the SDS index (P-value = 0.321). However, there are notable distinctions in the RAS (P-value = 0.0001) and MID (P-value = 0.0002) indices. Subsequent paired t-test results demonstrate that the MID of the DPTMOHS algorithm outperforms that of NSGA-II. Conversely, the RAS of the NSGA-II algorithm surpasses that of the DPTMOHS algorithm.

As outlined, the probability of failure for any VC deal is very high. Therefore, we propose a new approach for the first

Table 3: Estimation of startup cash flow.

Startup	Probability function	Parameters	Cash flow					
			Zero year	First year	Second year	Third year	Fourth year	Fifth year
Deal 1	Uniform	Worst case	−30	0.66	0.75	0.8	0.83	0.85
		Best case		3	12	20	30	42
Deal 2	Uniform	Worst case	−35	0.5	1.52	1.75	1.4	1.2
		Best case		2	10.5	28	35	42
Deal 3	Uniform	Worst case	−50	2	2	2	1.8	2
		Best case		8	18	32	45	72
Deal 4	Uniform	Worst case	−80	0	5.15	5.33	3.5	7.4
		Best case		18	26	48	48	185
Deal 5	Uniform	Worst case	−75	15	7	3.67	3.06	1.2
		Best case		60	63	60	50.3	43.2
Deal 6	Uniform	Worst case	−50	0.3	0.875	1.4	1.16	1
		Best case		3	14	35	42	49
Deal 7	Uniform	Worst case	−30	0.66	0.75	0.8	0.83	0.85
		Best case		3	12	20	30	42
Deal 8	Uniform	Worst case	−40	1.3	1.5	1.6	1.5	1.71
		Best case		12	24	40	54	84
Deal 9	Uniform	Worst case	−95	3	3.25	3.2	2.3	5.2
		Best case		18	52	80	84	259
Deal 10	Uniform	Worst case	−94	10	5.25	2.94	1.71	1.02
		Best case		90	48	73.5	61.8	50.4

Table 4: Strategy misalignment of each candidate.

	Deal 1	Deal 2	Deal 3	Deal 4	Deal 5	Deal 6	Deal 7	Deal 8	Deal 9	Deal 10
Strategy misalignment	5	2	2	1	4	3	2	1	2	4

objective of portfolio selection, which involves replacing the right tail of the probability function. This involves a utility function based on the probability of the right tail of PDNPV relative to the probability of negative PDNPV (see Equation (21)).

$$\frac{\left(\begin{array}{c}\sum_{j=1}^{k}\sum_{i=1}^{n_j}\sum_{t=1}^{T_{ji}}\Phi_{jiu}(\mathrm{MNPV})\alpha_{jiu}+\\ \sum_{j=1}^{k}\sum_{i=1}^{n_j}\int_{-\infty}^{\mathrm{MNPV}}\left(\Phi_{jiu}(\mathrm{MNPV}-h)\cdot\chi_{jiu}\right)f^{\left(\chi_{1,1,1},\ldots,\chi_{k,n_j,T_{ji}}\right)}(h)\,dh\end{array}\right)}{\left(1-\begin{array}{c}\sum_{j=1}^{k}\sum_{i=1}^{n_j}\sum_{t=1}^{T_{ji}}\Phi_{jiu}(\mathrm{MNPV})\alpha_{jiu}+\\ \sum_{j=1}^{k}\sum_{i=1}^{n_j}\int_{-\infty}^{0}\left(\Phi_{jiu}(0-h)\cdot\chi_{jiu}\right)f^{\left(\chi_{1,1,1},\ldots,\chi_{k,n_j,T_{ji}}\right)}(h)\,dh\end{array}\right)}. \tag{21}$$

To facilitate a comparison between the two approaches, we have designed 20 test problems (refer to Table 14). After implementing the DPTMOHS algorithm for each of these test problems, the efficiency of each approach is assessed through the proposed stress test, known as the deal failure rate. In this test, we assume that the NPV of each selected deal in the portfolio fails randomly (using Monte Carlo simulation). The failure is determined by selecting from the negative area of the NPV distribution function for each deal.

To assess whether there is a significant difference between the two approaches, the Paired Wilcoxon Rank test (in this test, there is no assumption of normal data distribution) is employed with a confidence interval of 0.95 (P-value = 0.00001) [67]. The results show a significant difference between the two methods, as illustrated in Table 9.

5.3. Discussion. As pointed out earlier, both the value of a business's cash flows and the contractual terms have a significant impact on the benefit to an investor when considering a startup. In this research, we illustrate how to evaluate a combination of specific contractual terms. Utilizing RO theory, decision trees, and PDNPV, we propose a practical formulation for valuing call options, liquidity preferences, and participant rights in VC contracts. We then present a practical mathematical model for selecting a combination of a deal portfolio based on the right tail of PDNPV and strategy alignment. To solve the model, we compare the performance of NSGA-II, DPTMOHS, and MBHS algorithms. The results of statistical hypothesis tests indicate no significant difference

TABLE 5: Cumulative probability distribution of NPV.

Proba-bility	Cumulative distribution function of net present value																																																	
	Deal 1	Deal 1−1	Deal 1−2	Deal 1−3	Deal 1−4	Deal 2	Deal 2−1	Deal 2−2	Deal 2−3	Deal 2−4	Deal 3	Deal 3−1	Deal 3−2	Deal 3−3	Deal 3−4	Deal 4	Deal 4−1	Deal 4−2	Deal 4−3	Deal 4−4	Deal 5	Deal 5−1	Deal 5−2	Deal 5−3	Deal 5−4	Deal 6	Deal 6−1	Deal 6−2	Deal 6−3	Deal 6−4	Deal 7	Deal 7−1	Deal 7−2	Deal 7−3	Deal 7−4	Deal 8	Deal 8−1	Deal 8−2	Deal 8−3	Deal 8−4	Deal 9	Deal 9−1	Deal 9−2	Deal 9−3	Deal 9−4	Deal 10	Deal 10−1	Deal 10−2	Deal 10−3	Deal 10−4
1%	−30	−39	−39	−37	−36	−33	−37	−36	−33	−33	−50	−77	−79	−75	−71	−79	−120	−118	−111	−107	−72	−115	−112	−105	−101	−40	−26	−25	−23	−24	−27	−31	−30	−27	−27	−22	−27	−27	−27	−26	−92	−162	−154	−139	−138	−84	−130	−125	−115	−115
2%	−29	−37	−36	−34	−32	−33	−34	−33	−30	−30	−50	−74	−75	−69	−64	−79	−113	−110	−101	−99	−71	−109	−105	−99	−94	−39	−24	−22	−21	−22	−27	−29	−27	−25	−24	−19	−25	−25	−25	−24	−91	−153	−145	−131	−128	−83	−121	−117	−107	−105
3%	−29	−36	−34	−32	−30	−32	−33	−31	−28	−28	−50	−71	−72	−65	−58	−78	−108	−106	−96	−91	−70	−105	−100	−94	−88	−38	−21	−20	−19	−20	−26	−27	−25	−23	−23	−18	−24	−24	−24	−23	−90	−147	−139	−124	−120	−81	−115	−111	−101	−99
4%	−29	−35	−33	−30	−28	−32	−31	−29	−26	−26	−50	−70	−70	−62	−53	−78	−105	−101	−91	−85	−69	−101	−96	−89	−84	−37	−19	−19	−18	−19	−26	−26	−24	−22	−21	−16	−23	−23	−23	−22	−89	−144	−133	−119	−113	−80	−111	−106	−95	−95
5%	−29	−33	−32	−29	−27	−32	−30	−28	−25	−25	−50	−68	−67	−59	−50	−78	−102	−97	−86	−80	−69	−98	−93	−85	−82	−37	−18	−18	−17	−18	−26	−25	−23	−21	−21	−15	−23	−22	−22	−22	−88	−139	−128	−115	−109	−80	−107	−101	−91	−90
6%	−29	−32	−30	−27	−26	−31	−29	−27	−24	−24	−50	−66	−65	−57	−50	−77	−99	−94	−82	−80	−68	−95	−90	−82	−78	−36	−17	−17	−16	−17	−25	−24	−22	−20	−20	−14	−22	−22	−21	−21	−87	−135	−124	−111	−105	−79	−104	−97	−89	−90
7%	−29	−31	−29	−26	−25	−31	−28	−26	−23	−23	−50	−65	−63	−53	−50	−77	−97	−90	−80	−80	−67	−93	−87	−80	−76	−36	−15	−15	−15	−16	−25	−23	−21	−19	−19	−13	−21	−21	−21	−21	−86	−132	−120	−107	−101	−78	−101	−94	−87	−89
8%	−29	−31	−28	−25	−25	−31	−26	−24	−22	−22	−50	−64	−61	−50	−50	−77	−94	−87	−78	−78	−67	−90	−85	−77	−75	−35	−13	−15	−15	−15	−25	−22	−20	−18	−19	−13	−21	−21	−20	−20	−86	−128	−116	−104	−98	−77	−98	−91	−84	−87
9%	−29	−30	−27	−24	−24	−30	−25	−23	−21	−21	−50	−62	−59	−50	−49	−76	−92	−84	−75	−77	−66	−88	−82	−75	−75	−35	−12	−14	−14	−15	−25	−21	−19	−18	−18	−12	−21	−20	−20	−20	−85	−125	−112	−100	−95	−76	−96	−89	−82	−85
10%	−29	−29	−26	−24	−23	−30	−24	−22	−20	−21	−50	−61	−56	−49	−48	−76	−89	−82	−74	−75	−65	−87	−80	−75	−75	−34	−11	−13	−13	−14	−24	−20	−18	−17	−18	−11	−20	−20	−20	−19	−84	−122	−109	−96	−95	−75	−93	−87	−80	−83
11%	−28	−28	−25	−23	−23	−30	−23	−21	−20	−20	−50	−60	−54	−47	−47	−76	−87	−79	−72	−74	−65	−85	−77	−74	−75	−34	−9	−12	−13	−14	−24	−20	−18	−17	−17	−11	−20	−19	−19	−19	−84	−120	−106	−95	−95	−74	−91	−84	−78	−81
12%	−28	−28	−24	−22	−22	−30	−23	−20	−19	−20	−50	−59	−52	−47	−47	−75	−85	−77	−70	−73	−64	−83	−75	−73	−73	−33	−8	−11	−12	−13	−24	−19	−17	−16	−17	−10	−19	−19	−19	−19	−83	−116	−103	−95	−95	−74	−88	−82	−76	−79
13%	−28	−27	−23	−22	−22	−29	−21	−19	−19	−19	−50	−58	−50	−46	−46	−75	−82	−74	−69	−71	−64	−82	−74	−71	−72	−33	−7	−10	−11	−12	−23	−18	−16	−16	−16	−10	−19	−19	−19	−18	−82	−114	−100	−95	−95	−73	−86	−80	−74	−78
14%	−28	−26	−22	−21	−22	−29	−20	−18	−18	−19	−50	−56	−48	−45	−46	−75	−80	−72	−68	−70	−63	−80	−73	−70	−71	−33	−6	−10	−11	−12	−23	−17	−16	−15	−16	−9	−19	−18	−18	−18	−82	−111	−97	−95	−95	−72	−84	−78	−73	−76
15%	−28	−25	−21	−21	−21	−29	−19	−18	−17	−18	−50	−55	−46	−44	−45	−75	−78	−70	−67	−69	−62	−78	−71	−68	−70	−32	−5	−9	−10	−12	−23	−16	−15	−15	−16	−9	−19	−18	−18	−18	−81	−109	−95	−95	−95	−71	−82	−76	−72	−75
16%	−28	−24	−20	−20	−21	−29	−18	−17	−17	−18	−49	−54	−44	−43	−44	−74	−76	−68	−66	−68	−62	−77	−70	−67	−69	−32	−4	−8	−10	−11	−23	−16	−15	−15	−15	−8	−18	−18	−18	−18	−80	−107	−95	−95	−95	−70	−80	−74	−70	−73
17%	−28	−23	−20	−20	−21	−28	−17	−16	−17	−18	−49	−52	−43	−42	−44	−74	−74	−66	−65	−67	−61	−75	−68	−66	−68	−32	−4	−7	−9	−11	−23	−15	−14	−14	−15	−8	−18	−18	−18	−17	−80	−104	−95	−93	−95	−69	−79	−73	−69	−72
18%	−28	−22	−19	−20	−20	−28	−16	−16	−16	−17	−49	−50	−42	−42	−43	−74	−72	−64	−64	−66	−61	−74	−66	−65	−67	−31	−3	−7	−9	−10	−22	−14	−14	−14	−15	−7	−18	−17	−17	−17	−79	−102	−94	−92	−95	−69	−77	−71	−68	−71
19%	−28	−20	−18	−19	−20	−28	−15	−15	−16	−17	−49	−47	−40	−41	−43	−73	−69	−63	−63	−65	−60	−72	−65	−64	−66	−31	−3	−6	−8	−10	−22	−13	−13	−13	−14	−7	−17	−17	−17	−17	−78	−100	−92	−90	−93	−68	−75	−69	−66	−70
20%	−28	−19	−18	−19	−20	−28	−14	−15	−15	−17	−49	−44	−39	−41	−42	−73	−66	−62	−62	−65	−60	−70	−64	−63	−65	−31	−2	−6	−8	−9	−22	−13	−13	−13	−14	−7	−17	−17	−17	−17	−78	−97	−90	−88	−92	−67	−73	−68	−65	−69
21%	−27	−18	−17	−18	−20	−28	−13	−14	−15	−16	−49	−41	−38	−40	−42	−73	−64	−60	−61	−64	−59	−69	−63	−62	−64	−30	−2	−5	−8	−9	−22	−12	−12	−13	−14	−6	−17	−17	−17	−16	−77	−95	−88	−87	−90	−66	−71	−66	−64	−67
22%	−27	−17	−17	−18	−19	−27	−12	−14	−15	−16	−49	−38	−38	−40	−41	−72	−61	−59	−60	−63	−58	−68	−62	−61	−63	−30	−2	−5	−7	−9	−21	−11	−12	−13	−13	−6	−17	−16	−16	−16	−76	−93	−87	−86	−89	−66	−70	−65	−63	−66
23%	−27	−16	−17	−18	−19	−27	−12	−13	−14	−16	−49	−35	−37	−39	−41	−72	−59	−58	−59	−62	−58	−66	−60	−60	−63	−30	−2	−4	−7	−8	−21	−11	−11	−12	−13	−5	−16	−16	−16	−16	−75	−91	−85	−84	−88	−65	−68	−64	−62	−65
24%	−27	−15	−16	−18	−19	−27	−11	−13	−14	−15	−49	−34	−36	−39	−41	−72	−56	−57	−58	−61	−57	−65	−59	−59	−62	−29	−1	−4	−7	−8	−21	−10	−11	−12	−13	−5	−16	−16	−16	−16	−75	−90	−83	−83	−86	−64	−66	−63	−60	−64
25%	−27	−15	−16	−17	−18	−27	−11	−13	−14	−15	−49	−33	−36	−38	−40	−71	−54	−56	−57	−60	−57	−63	−58	−59	−61	−29	−1	−4	−6	−8	−21	−10	−11	−12	−13	−5	−16	−16	−16	−16	−74	−88	−82	−82	−85	−63	−65	−61	−59	−63
26%	−27	−14	−16	−17	−18	−26	−10	−12	−14	−15	−49	−31	−35	−38	−40	−71	−53	−54	−56	−59	−56	−62	−57	−58	−61	−29	−1	−4	−6	−7	−21	−9	−10	−11	−12	−4	−16	−16	−16	−15	−73	−86	−80	−80	−84	−62	−63	−60	−58	−62
27%	−27	−13	−15	−17	−18	−26	−10	−12	−13	−15	−49	−30	−34	−37	−39	−70	−51	−53	−55	−59	−55	−60	−56	−57	−60	−29	−1	−3	−6	−7	−20	−9	−10	−11	−12	−4	−16	−15	−15	−15	−72	−84	−79	−79	−82	−61	−60	−59	−57	−62
28%	−27	−13	−15	−16	−18	−26	−9	−12	−13	−14	−49	−30	−34	−37	−39	−70	−50	−52	−55	−58	−54	−59	−55	−56	−59	−28	−1	−3	−5	−7	−20	−8	−10	−11	−12	−4	−15	−15	−15	−15	−71	−82	−77	−78	−81	−61	−59	−58	−57	−61
29%	−26	−13	−15	−16	−18	−26	−9	−11	−13	−14	−49	−29	−33	−36	−39	−70	−48	−51	−54	−57	−54	−57	−54	−55	−58	−28	−1	−3	−5	−7	−20	−8	−10	−11	−12	−3	−15	−15	−15	−15	−71	−80	−76	−77	−80	−60	−57	−56	−56	−60
30%	−26	−12	−14	−16	−17	−25	−9	−11	−12	−14	−49	−28	−33	−36	−38	−69	−47	−50	−53	−57	−53	−56	−53	−55	−58	−28	−1	−3	−5	−6	−20	−8	−9	−10	−11	−3	−15	−15	−15	−15	−70	−79	−75	−76	−79	−59	−56	−55	−55	−59
31%	−26	−12	−14	−16	−17	−25	−8	−11	−12	−14	−49	−27	−33	−36	−38	−69	−45	−50	−52	−56	−53	−54	−52	−54	−57	−28	−1	−2	−4	−6	−19	−7	−9	−10	−11	−2	−15	−15	−15	−14	−69	−76	−73	−75	−77	−58	−54	−54	−54	−58
32%	−26	−11	−14	−16	−17	−25	−8	−10	−12	−13	−49	−27	−32	−35	−38	−68	−44	−49	−51	−55	−52	−53	−51	−53	−56	−27	−1	−2	−4	−6	−19	−7	−9	−10	−11	−2	−15	−15	−15	−14	−68	−75	−72	−74	−76	−57	−52	−53	−53	−57
33%	−26	−11	−14	−15	−17	−25	−8	−10	−12	−13	−49	−26	−32	−35	−37	−68	−43	−48	−51	−54	−51	−51	−50	−52	−56	−27	−1	−2	−4	−6	−19	−7	−8	−10	−11	−2	−14	−14	−14	−14	−68	−73	−70	−73	−75	−56	−51	−52	−52	−56
34%	−26	−11	−13	−15	−16	−24	−7	−10	−11	−13	−48	−26	−31	−34	−37	−67	−42	−47	−50	−54	−51	−50	−49	−52	−55	−27	−1	−2	−4	−5	−19	−6	−8	−10	−11	−2	−14	−14	−14	−14	−67	−71	−69	−71	−74	−55	−50	−51	−51	−55
35%	−25	−10	−13	−15	−16	−24	−7	−10	−11	−13	−48	−25	−31	−34	−37	−67	−40	−46	−49	−53	−50	−49	−49	−51	−55	−27	0	−2	−4	−5	−18	−6	−8	−9	−10	−1	−14	−14	−14	−14	−66	−69	−67	−70	−73	−55	−48	−49	−51	−55
36%	−25	−10	−13	−15	−16	−24	−7	−9	−11	−12	−48	−24	−30	−34	−36	−66	−39	−45	−48	−52	−50	−47	−48	−50	−54	−26	0	−2	−3	−5	−18	−6	−8	−9	−10	−1	−14	−14	−14	−14	−65	−68	−66	−69	−72	−54	−46	−48	−50	−54
37%	−25	−10	−13	−15	−16	−24	−7	−9	−11	−12	−48	−24	−30	−33	−36	−66	−38	−44	−47	−51	−49	−46	−47	−50	−53	−26	0	−2	−3	−5	−18	−6	−8	−9	−10	0	−14	−14	−14	−13	−63	−66	−65	−68	−71	−53	−45	−48	−49	−53
38%	−25	−10	−12	−14	−16	−23	−6	−9	−11	−12	−48	−24	−30	−33	−36	−65	−37	−43	−47	−51	−48	−45	−46	−49	−53	−26	0	−1	−3	−4	−18	−5	−7	−9	−10	0	−14	−14	−13	−13	−62	−64	−64	−66	−69	−52	−43	−47	−48	−52
39%	−25	−9	−12	−14	−15	−23	−6	−9	−10	−12	−48	−23	−29	−33	−35	−65	−36	−42	−46	−50	−47	−44	−45	−48	−52	−25	0	−1	−3	−4	−18	−5	−7	−9	−10	0	−13	−13	−13	−13	−62	−62	−62	−65	−68	−51	−42	−46	−48	−51
40%	−25	−9	−12	−14	−15	−23	−6	−8	−10	−12	−48	−23	−29	−32	−35	−64	−35	−42	−45	−49	−46	−43	−45	−48	−51	−25	0	−1	−3	−4	−17	−5	−7	−8	−9	1	−13	−13	−13	−13	−61	−61	−61	−64	−67	−50	−41	−45	−47	−51
41%	−24	−9	−12	−14	−15	−23	−6	−8	−10	−11	−48	−22	−28	−32	−35	−64	−34	−41	−45	−49	−46	−42	−44	−47	−51	−25	0	−1	−3	−4	−17	−5	−7	−8	−9	1	−13	−13	−13	−13	−60	−59	−60	−63	−67	−49	−39	−44	−46	−50
42%	−24	−9	−11	−14	−15	−22	−5	−8	−10	−11	−48	−22	−28	−32	−35	−63	−33	−40	−44	−48	−45	−41	−43	−46	−50	−25	0	−1	−2	−4	−17	−5	−7	−8	−9	1	−13	−13	−13	−13	−59	−57	−58	−62	−65	−48	−38	−43	−45	−49
43%	−24	−8	−11	−13	−15	−22	−5	−8	−10	−11	−48	−22	−28	−31	−34	−62	−32	−39	−43	−47	−44	−40	−43	−46	−50	−24	0	−1	−2	−3	−17	−4	−6	−8	−9	2	−13	−13	−13	−12	−57	−56	−57	−61	−64	−47	−37	−42	−44	−48

TABLE 5: Continued.

Probability	Cumulative distribution function of net present value																																																	
	Deal 1	Deal 1−1	Deal 1−2	Deal 1−3	Deal 1−4	Deal 2	Deal 2−1	Deal 2−2	Deal 2−3	Deal 2−4	Deal 3	Deal 3−1	Deal 3−2	Deal 3−3	Deal 3−4	Deal 4	Deal 4−1	Deal 4−2	Deal 4−3	Deal 4−4	Deal 5	Deal 5−1	Deal 5−2	Deal 5−3	Deal 5−4	Deal 6	Deal 6−1	Deal 6−2	Deal 6−3	Deal 6−4	Deal 7	Deal 7−1	Deal 7−2	Deal 7−3	Deal 7−4	Deal 8	Deal 8−1	Deal 8−2	Deal 8−3	Deal 8−4	Deal 9	Deal 9−1	Deal 9−2	Deal 9−3	Deal 9−4	Deal 10	Deal 10−1	Deal 10−2	Deal 10−3	Deal 10−4
44%	−24	−8	−11	−13	−14	−22	−5	−8	−9	−11	−47	−21	−28	−31	−34	−62	−31	−38	−43	−47	−43	−39	−42	−45	−49	−24	0	−1	−2	−3	−16	−4	−6	−8	−9	2	−13	−13	−12	−12	−56	−54	−55	−59	−63	−46	−36	−41	−44	−47
45%	−24	−8	−11	−13	−14	−21	−5	−7	−9	−11	−47	−21	−27	−31	−34	−61	−30	−38	−42	−46	−43	−38	−42	−45	−49	−24	0	−1	−2	−3	−16	−4	−6	−7	−8	2	−12	−12	−12	−12	−55	−53	−54	−59	−62	−45	−35	−40	−43	−47
46%	−23	−8	−11	−13	−14	−21	−5	−7	−9	−10	−47	−21	−27	−30	−33	−61	−29	−37	−41	−45	−42	−37	−41	−44	−48	−24	0	−1	−2	−3	−16	−4	−6	−7	−8	3	−12	−12	−12	−12	−54	−51	−53	−58	−61	−44	−34	−39	−42	−46
47%	−23	−8	−10	−13	−14	−21	−5	−7	−9	−10	−47	−20	−26	−30	−33	−60	−28	−36	−40	−45	−41	−36	−40	−43	−47	−23	0	−1	−2	−3	−15	−4	−6	−7	−8	3	−12	−12	−12	−12	−53	−50	−52	−57	−60	−42	−32	−38	−41	−45
48%	−23	−7	−10	−12	−14	−21	−4	−7	−9	−10	−47	−20	−26	−30	−33	−59	−28	−35	−40	−44	−40	−36	−39	−43	−47	−23	0	−1	−2	−3	−15	−4	−6	−7	−8	3	−12	−12	−12	−12	−52	−48	−51	−56	−59	−41	−31	−38	−40	−44
49%	−23	−7	−10	−12	−13	−20	−4	−7	−8	−10	−47	−20	−26	−29	−32	−59	−27	−35	−39	−43	−39	−35	−39	−42	−46	−23	0	−1	−2	−3	−15	−3	−5	−7	−8	4	−12	−12	−12	−11	−50	−46	−50	−55	−58	−40	−30	−37	−40	−44
50%	−22	−7	−10	−12	−13	−20	−4	−7	−8	−10	−47	−19	−26	−29	−32	−58	−26	−34	−38	−43	−38	−34	−38	−41	−45	−23	0	−1	−2	−3	−15	−3	−5	−7	−8	4	−12	−12	−12	−11	−49	−45	−48	−54	−57	−39	−29	−36	−39	−43
51%	−22	−7	−10	−12	−13	−19	−4	−6	−8	−9	−46	−19	−25	−29	−32	−57	−25	−33	−37	−42	−37	−33	−37	−41	−45	−22	0	−1	−1	−2	−14	−3	−5	−6	−7	4	−11	−11	−11	−11	−48	−44	−47	−53	−56	−38	−28	−35	−38	−42
52%	−22	−7	−10	−12	−13	−19	−4	−6	−8	−9	−46	−19	−25	−28	−31	−56	−25	−33	−37	−41	−36	−32	−37	−40	−44	−22	0	0	−1	−2	−14	−3	−5	−6	−7	5	−11	−11	−11	−11	−46	−42	−46	−52	−55	−37	−27	−34	−37	−41
53%	−22	−7	−9	−11	−13	−19	−4	−6	−8	−9	−46	−18	−25	−28	−31	−56	−24	−32	−36	−41	−35	−31	−36	−40	−43	−22	0	0	−1	−2	−14	−3	−5	−6	−7	5	−11	−11	−11	−11	−45	−40	−45	−50	−54	−35	−26	−33	−36	−40
54%	−21	−6	−9	−11	−13	−18	−4	−6	−8	−9	−46	−18	−24	−28	−31	−55	−23	−31	−35	−40	−34	−31	−35	−39	−43	−22	0	0	−1	−2	−13	−3	−5	−6	−7	5	−11	−11	−11	−11	−44	−39	−44	−49	−53	−34	−26	−32	−36	−40
55%	−21	−6	−9	−11	−12	−18	−3	−6	−7	−9	−46	−18	−24	−28	−31	−54	−22	−30	−35	−39	−33	−30	−35	−38	−42	−21	0	0	−1	−2	−13	−3	−4	−6	−7	6	−11	−11	−11	−11	−42	−38	−43	−48	−52	−32	−25	−31	−35	−39
56%	−21	−6	−9	−11	−12	−18	−3	−6	−7	−9	−45	−17	−24	−27	−30	−53	−22	−30	−34	−39	−31	−29	−34	−38	−42	−21	0	0	−1	−2	−13	−2	−4	−6	−6	6	−11	−11	−11	−10	−40	−36	−42	−47	−51	−31	−24	−31	−34	−38
57%	−21	−6	−9	−11	−12	−17	−3	−5	−7	−8	−45	−17	−23	−27	−30	−52	−21	−29	−33	−38	−30	−28	−33	−37	−41	−21	0	0	−1	−2	−12	−2	−4	−5	−6	6	−10	−10	−10	−10	−38	−35	−41	−46	−51	−29	−23	−30	−33	−37
58%	−20	−6	−9	−11	−12	−17	−3	−5	−7	−8	−45	−17	−23	−27	−30	−51	−20	−28	−33	−37	−29	−28	−33	−37	−40	−20	0	0	−1	−2	−12	−2	−4	−5	−6	7	−10	−10	−10	−10	−37	−34	−40	−45	−49	−28	−22	−29	−33	−36
59%	−20	−6	−8	−10	−12	−17	−3	−5	−7	−8	−45	−16	−23	−26	−29	−50	−19	−28	−32	−37	−28	−27	−32	−36	−40	−20	0	0	−1	−2	−12	−2	−4	−5	−6	7	−10	−10	−10	−10	−35	−32	−39	−44	−48	−26	−21	−28	−32	−36
60%	−19	−6	−8	−10	−11	−16	−3	−5	−7	−8	−44	−16	−23	−26	−29	−49	−19	−27	−31	−36	−26	−26	−31	−35	−39	−20	0	0	−1	−1	−11	−2	−4	−5	−6	7	−10	−10	−10	−10	−33	−31	−38	−43	−47	−25	−20	−27	−31	−35
61%	−19	−5	−8	−10	−11	−16	−3	−5	−6	−8	−44	−16	−22	−26	−29	−48	−18	−26	−31	−36	−25	−25	−31	−35	−39	−19	0	0	−1	−1	−11	−2	−4	−5	−6	8	−10	−10	−10	−10	−31	−30	−37	−42	−46	−23	−20	−27	−31	−34
62%	−19	−5	−8	−10	−11	−15	−3	−5	−6	−7	−44	−16	−22	−25	−28	−47	−18	−26	−30	−35	−23	−24	−30	−34	−38	−19	0	0	−1	−1	−11	−2	−3	−5	−6	8	−10	−9	−9	−10	−29	−29	−36	−41	−45	−22	−19	−26	−30	−33
63%	−18	−5	−8	−10	−11	−15	−2	−4	−6	−7	−43	−15	−22	−25	−28	−46	−17	−25	−30	−34	−22	−24	−29	−33	−37	−19	0	0	−1	−1	−10	−2	−3	−5	−5	8	−9	−9	−9	−9	−27	−27	−34	−39	−44	−20	−18	−25	−29	−33
64%	−18	−5	−7	−9	−11	−14	−2	−4	−6	−7	−43	−15	−21	−25	−28	−45	−16	−24	−29	−34	−21	−23	−28	−33	−37	−19	0	0	−1	−1	−10	−2	−3	−4	−5	9	−9	−9	−9	−9	−25	−26	−33	−38	−43	−18	−17	−24	−28	−32
65%	−18	−5	−7	−9	−11	−14	−2	−4	−6	−7	−43	−15	−21	−24	−28	−43	−16	−24	−28	−33	−19	−22	−28	−32	−36	−18	0	0	0	−1	−9	−2	−3	−4	−5	9	−9	−9	−9	−9	−23	−25	−32	−37	−42	−17	−17	−23	−27	−31
66%	−17	−5	−7	−9	−10	−13	−2	−4	−5	−7	−42	−14	−21	−24	−27	−42	−15	−23	−28	−32	−17	−21	−27	−31	−35	−18	0	0	0	−1	−9	−1	−3	−4	−5	9	−9	−9	−9	−9	−21	−24	−31	−36	−41	−15	−16	−22	−26	−30
67%	−17	−4	−7	−9	−10	−13	−2	−4	−5	−6	−42	−14	−20	−24	−27	−41	−15	−22	−27	−31	−16	−21	−26	−31	−34	−18	0	0	0	−1	−8	−1	−3	−4	−5	10	−8	−9	−9	−9	−18	−23	−30	−35	−40	−13	−15	−22	−26	−29
68%	−16	−4	−7	−9	−10	−12	−2	−4	−5	−6	−41	−14	−20	−23	−27	−39	−14	−22	−26	−31	−14	−20	−25	−30	−33	−17	0	0	0	−1	−8	−1	−3	−4	−4	10	−8	−8	−8	−8	−16	−22	−29	−34	−39	−11	−14	−21	−25	−29
69%	−16	−4	−7	−8	−10	−12	−2	−4	−5	−6	−41	−14	−20	−23	−26	−37	−13	−21	−26	−30	−12	−19	−25	−29	−33	−17	0	0	0	−1	−8	−1	−2	−4	−4	10	−8	−8	−8	−8	−13	−21	−28	−33	−37	−9	−13	−20	−24	−28
70%	−15	−4	−6	−8	−10	−11	−2	−3	−5	−6	−40	−13	−19	−23	−26	−36	−13	−20	−25	−29	−10	−18	−24	−28	−32	−17	0	0	0	−1	−7	−1	−2	−3	−4	11	−8	−8	−8	−8	−11	−20	−27	−32	−36	−7	−13	−19	−23	−27
71%	−15	−4	−6	−8	−9	−10	−2	−3	−5	−6	−40	−13	−19	−22	−26	−34	−12	−20	−24	−29	−8	−18	−23	−27	−31	−16	0	0	0	−1	−7	−1	−2	−3	−4	11	−8	−8	−8	−8	−8	−19	−25	−31	−35	−5	−12	−18	−22	−26
72%	−14	−4	−6	−8	−9	−10	−1	−3	−4	−5	−39	−13	−19	−22	−25	−32	−12	−19	−24	−28	−6	−17	−22	−27	−30	−16	0	0	0	−1	−6	−1	−2	−3	−4	12	−7	−8	−8	−8	−6	−18	−24	−29	−34	−2	−11	−18	−21	−25
73%	−13	−4	−6	−8	−9	−9	−1	−3	−4	−5	−38	−12	−18	−22	−25	−30	−11	−18	−23	−27	−4	−16	−22	−26	−29	−15	0	0	0	−1	−6	−1	−2	−3	−4	12	−7	−7	−7	−7	−3	−16	−23	−28	−32	1	−10	−17	−20	−24
74%	−13	−3	−6	−7	−9	−8	−1	−3	−4	−5	−38	−12	−18	−21	−24	−27	−11	−18	−22	−26	−2	−15	−21	−25	−28	−15	0	0	0	0	−5	−1	−2	−3	−3	12	−7	−7	−7	−7	0	−15	−22	−27	−31	3	−10	−16	−19	−23
75%	−12	−3	−6	−7	−8	−8	−1	−3	−4	−5	−37	−12	−18	−21	−24	−25	−10	−17	−22	−26	1	−15	−20	−24	−27	−15	0	0	0	0	−4	−1	−2	−3	−3	13	−7	−7	−7	−7	3	−14	−21	−25	−29	6	−9	−15	−18	−22
76%	−11	−3	−5	−7	−8	−7	−1	−3	−4	−5	−36	−11	−17	−21	−24	−23	−10	−17	−21	−25	3	−14	−19	−22	−26	−14	0	0	0	0	−4	−1	−2	−2	−3	13	−6	−7	−7	−7	7	−13	−19	−24	−28	9	−8	−14	−17	−20
77%	−10	−3	−5	−7	−8	−6	−1	−2	−3	−4	−35	−11	−17	−20	−23	−20	−9	−16	−20	−24	6	−13	−18	−21	−25	−14	0	0	0	0	−3	−1	−1	−2	−3	14	−6	−6	−7	−7	11	−12	−18	−22	−26	12	−7	−13	−16	−19
78%	−10	−3	−5	−6	−8	−5	−1	−2	−3	−4	−34	−11	−17	−20	−23	−17	−9	−15	−19	−23	9	−12	−17	−20	−24	−13	0	0	0	0	−2	−1	−1	−2	−3	14	−6	−6	−6	−6	15	−11	−16	−21	−25	14	−7	−12	−14	−18
79%	−9	−3	−5	−6	−7	−4	−1	−2	−3	−4	−33	−11	−16	−20	−22	−14	−8	−15	−18	−23	12	−11	−16	−19	−22	−13	0	0	0	0	−2	0	−1	−2	−2	15	−5	−6	−6	−6	19	−9	−15	−19	−23	18	−6	−11	−13	−17
80%	−8	−2	−4	−6	−7	−3	−1	−2	−3	−4	−32	−10	−16	−19	−22	−11	−8	−14	−17	−22	16	−10	−14	−17	−20	−12	0	0	0	0	−1	0	−1	−2	−2	15	−5	−6	−6	−6	24	−8	−14	−17	−21	22	−5	−10	−12	−15
81%	−7	−2	−4	−6	−7	−2	−1	−2	−2	−3	−30	−10	−15	−19	−21	−7	−7	−13	−16	−21	19	−9	−13	−16	−18	−12	0	0	0	0	0	0	−1	−2	−2	16	−5	−5	−5	−6	29	−7	−12	−15	−19	25	−4	−9	−9	−13
82%	−5	−2	−4	−5	−6	−1	−1	−2	−2	−3	−29	−9	−15	−18	−21	−2	−7	−13	−16	−20	23	−8	−11	−14	−16	−11	0	0	0	0	1	0	−1	−1	−2	16	−4	−5	−5	−5	35	−6	−11	−13	−17	29	−3	−8	−7	−11
83%	−4	−2	−4	−5	−6	0	0	−1	−2	−3	−27	−9	−15	−18	−21	2	−6	−12	−15	−19	27	−7	−9	−11	−13	−11	0	0	0	0	2	0	−1	−1	−1	17	−4	−5	−5	−5	41	−5	−9	−11	−14	33	−2	−6	−4	−9
84%	−3	−2	−3	−5	−6	2	0	−1	−2	−2	−25	−9	−14	−17	−20	6	−6	−11	−14	−18	32	−5	−6	−8	−10	−10	0	0	0	0	3	0	0	−1	−1	17	−3	−4	−4	−5	47	−3	−6	−8	−10	38	−1	−4	0	−5
85%	−1	−2	−3	−4	−5	3	0	−1	−1	−2	−23	−8	−14	−17	−19	12	−5	−10	−12	−17	38	−3	−1	−3	−5	−10	0	0	0	0	4	0	0	−1	−1	18	−3	−4	−4	−4	54	−2	−3	−4	−6	44	0	−1	5	0
86%	0	−1	−3	−4	−5	5	0	−1	−1	−1	−21	−8	−13	−16	−19	18	−4	−9	−11	−15	43	−1	4	2	0	−9	0	0	0	0	5	0	0	0	0	19	−2	−3	−4	−4	62	0	3	4	2	50	5	4	10	6

Table 5: Continued.

	Cumulative distribution function of net present value																																																	
Probability	Deal 1	Deal 1−1	Deal 1−2	Deal 1−3	Deal 1−4	Deal 2	Deal 2−1	Deal 2−2	Deal 2−3	Deal 2−4	Deal 3	Deal 3−1	Deal 3−2	Deal 3−3	Deal 3−4	Deal 4	Deal 4−1	Deal 4−2	Deal 4−3	Deal 4−4	Deal 5	Deal 5−1	Deal 5−2	Deal 5−3	Deal 5−4	Deal 6	Deal 6−1	Deal 6−2	Deal 6−3	Deal 6−4	Deal 7	Deal 7−1	Deal 7−2	Deal 7−3	Deal 7−4	Deal 8	Deal 8−1	Deal 8−2	Deal 8−3	Deal 8−4	Deal 9	Deal 9−1	Deal 9−2	Deal 9−3	Deal 9−4	Deal 10	Deal 10−1	Deal 10−2	Deal 10−3	Deal 10−4
87%	2	−1	−2	−3	−4	7	0	−1	−1	−1	−18	−8	−13	−16	−18	25	−4	−8	−9	−13	48	5	11	8	6	−8	0	0	0	0	6	0	0	0	1	19	−1	−3	−3	−4	69	8	11	13	11	55	11	11	16	12
88%	4	−1	−2	−3	−4	9	0	0	0	1	−14	−7	−12	−15	−17	32	−3	−7	−7	−10	56	12	17	14	12	−7	0	0	0	0	7	0	1	2	3	20	−1	−2	−3	−3	78	17	20	21	19	62	18	17	22	18
89%	6	−1	−2	−2	−3	11	0	0	3	3	−10	−7	−11	−14	−17	39	−2	−6	−2	−5	63	20	25	21	19	−6	0	0	0	0	9	0	3	4	4	21	0	−2	−2	−3	88	26	29	32	29	70	27	24	29	25
90%	9	−1	−1	−1	−1	13	0	2	5	5	−6	−6	−11	−14	−16	48	−1	−2	7	3	71	29	35	29	27	−6	0	0	0	0	11	2	5	5	5	22	0	−1	−2	−2	100	37	42	45	38	78	35	33	38	35
91%	12	0	1	2	2	16	1	5	8	8	−1	−6	−10	−13	−15	57	9	10	16	14	81	40	45	38	37	−5	0	0	0	0	13	4	7	7	7	22	0	−1	−1	−2	113	51	56	58	52	89	44	44	47	45
92%	16	0	4	6	6	19	4	8	11	11	6	−5	−9	−11	−14	68	19	20	30	29	93	50	57	47	48	−4	0	0	0	0	15	6	9	9	9	23	0	0	−1	−1	128	64	74	73	68	100	57	53	59	55
93%	20	3	8	10	12	23	7	11	14	14	14	−4	−8	−10	−12	83	33	34	44	44	106	64	70	60	60	−2	0	0	0	0	18	8	11	12	11	25	0	0	0	−1	146	85	91	90	88	114	71	64	73	65
94%	26	7	13	16	18	27	11	15	19	19	25	−3	−6	−7	−10	101	49	49	60	59	120	81	85	75	75	−1	0	0	0	0	20	12	15	15	14	26	0	0	0	−1	168	105	114	109	109	129	89	77	87	79
95%	34	13	22	23	24	31	16	20	23	24	38	−1	−2	3	−3	121	71	72	83	79	138	98	102	92	95	1	0	0	0	1	24	15	18	18	18	27	0	0	0	0	192	130	140	136	135	150	105	95	104	96
96%	43	23	33	34	35	37	23	27	29	30	60	13	17	24	18	147	100	101	109	109	171	129	128	117	129	3	0	1	2	3	29	20	22	22	22	29	0	0	0	0	231	168	178	178	179	177	132	118	132	112
97%	56	38	45	47	48	46	31	37	38	38	92	43	48	53	50	187	135	141	146	151	203	164	166	152	164	5	1	3	4	5	35	27	28	28	28	30	0	0	0	0	285	216	222	233	227	214	165	154	169	140
98%	76	60	65	66	71	60	44	50	52	51	156	93	101	106	102	253	202	202	210	209	267	225	223	205	214	9	5	5	7	8	46	38	36	38	34	33	1	0	1	1	359	292	297	312	317	263	215	203	219	200
99%	122	104	115	114	112	83	66	76	72	74	308	250	236	257	228	382	336	338	376	340	399	363	318	316	316	15	12	11	12	13	61	54	54	53	48	36	2	2	4	4	518	441	436	506	455	380	327	290	307	281
100%	942	934	763	805	935	320	266	396	450	305	5,417	5,389	2,494	3,565	4,436	3,034	3,001	2,835	3,455	2,554	2,318	2,283	2,820	1,567	1,684	52	47	44	48	45	275	264	242	254	205	60	22	15	24	41	2,602	2,546	5,858	3,413	2,582	1,324	1,242	1,850	1,166	1,108
Cost	**30**	**7**	**10**	**13**	**15**	**35**	**5**	**8**	**10**	**11**	**50**	**18**	**25**	**29**	**32**	**80**	**27**	**36**	**42**	**47**	**75**	**33**	**40**	**46**	**50**	**50**	**1**	**2**	**3**	**5**	**30**	**5**	**7**	**8**	**10**	**40**	**19**	**21**	**22**	**23**	**95**	**47**	**55**	**62**	**67**	**90**	**33**	**41**	**47**	**51**

Table 6: The percentage of finding Pareto solutions in algorithms.

Size	MNPV	Exact algorithm* (%)	DPTMOHS algorithm (%)
8	50	100	100
8	75	100	100
8	100	100	100
10	50	100	100
10	75	100	100
10	100	100	100
12	50	100	100
12	75	100	100
12	100	100	66

*This algorithm solves the problem by counting the entire feasible space.

Table 7: The Pareto solution of algorithms MOBDEHS for five iterations.

Solutions	Deal 1	Deal 2	Deal 3	Deal 4	Deal 5	Deal 6	Deal 7	Deal 8	Deal 9	Deal 10	Deal 11	Deal 12	Objective 1	Objective 2
1	0	0	0	0	0	1	1	1					0.073353	2
	1	0	0	0	0	1	1	1					0.11078383	2.75
	1	0	0	0	1	1	1	1					0.137424105	3.2
	1	0	1	0	1	1	0	1					0.145801031	3.8
	1	1	0	1	1	1	0	0					0.153424909	4.4
2	0	0	0	0	0	1	1	1					0.051881	2
	1	0	0	0	0	1	1	1					0.08084466	2.75
	1	0	0	0	1	1	1	1					0.100283076	3.2
	1	0	1	0	1	1	0	1					0.109298719	3.8
3	0	0	0	0	0	1	0	1					0.021	2
	0	1	1	0	0	0	0	0					0.0399	2.5
	0	0	0	1	1	0	0	0					0.0401	2.8333
	0	0	0	1	0	1	1	0					0.040931	3
	1	1	0	0	0	1	1	0					0.06071436	3.5
	1	1	0	0	1	1	0	0					0.07017149	4.25
4	0	0	0	0	0	1	0	1	1	1			0.11150892	2
	1	0	0	0	0	1	0	1	1	1			0.14675165	2.6
	1	0	0	1	0	1	0	0	1	1			0.154373822	3.2
5	0	0	0	0	0	1	0	1	1	1			0.06389883	2
	1	0	0	0	0	1	0	1	1	1			0.091650821	2.6
	1	0	0	0	1	1	0	1	0	1			0.100513936	3.2
	1	0	1	0	1	1	0	1	0	0			0.109298719	3.8
6	0	0	0	0	0	1	0	1	1	1			0.04275226	2
	1	0	0	0	0	1	0	1	1	1			0.062626546	2.6
	1	0	0	0	1	1	0	1	0	1			0.071408062	3.2
	1	0	1	0	1	1	0	1	0	0			0.080257478	3.8
	1	1	0	1	1	1	0	0	0	0			0.088969144	4.4
7	0	0	0	0	0	0	1	1	0	1	1	0	0.11936526	2
	1	0	0	0	0	1	0	1	1	1	0	0	0.14675165	2.6
	1	1	0	1	0	1	0	0	0	1	0	0	0.153449543	3.8
8	0	0	0	0	0	1	1	1	0	0	1	0	0.09078712	2
	1	0	0	0	0	1	0	1	1	1	0	0	0.091650821	2.6
	1	0	0	0	0	0	0	1	0	1	1	0	0.09976591	2.75
	1	1	0	0	0	0	1	1	0	0	1	0	0.117723069	3.2
9	0	0	0	0	0	1	1	1	0	0	1	0	0.06229335	2
	1	0	0	0	0	1	0	1	1	1	0	0	0.062626546	2.6
	1	0	0	0	0	0	0	1	1	0	1	0	0.07148536	2.75
	1	1	0	0	0	0	1	1	0	0	1	0	0.089625689	3.2

Table 8: Taguchi levels for MBHS.

Parameter	Level 1	Level 2	Level 3	Level 4
HMCR	0.7	0.9	0.95	0.99
PAR	0.1	0.3	0.4	0.5

Table 9: Taguchi levels for NSGA-II.

Parameter	Level 1	Level 2	Level 3
nPoP	20	30	40
Pcrossover	0.6	0.7	0.8
Pmutation	0.2	0.3	0.4

Table 10: The best level of MBHS algorithm.

Parameter	HMCR	PAR
Best level	0.95	0.4

Table 11: The best level of NSGA-II algorithm.

Parameter	nPoP (population size)	Pcrossover (crossover percentage)	Pmutation (mutation percentage)
Best level	20	0.7	0.4

Table 12: Comparing the performance of NSGA-II, DPTMOHS, and MBHS algorithms.

Test number	Size	Budgets	MNPV	NSGA-II			DPTMOHS			MBHS		
		B–E		MID	SDS	RAS	MID	SDS	RAS	MID	SDS	RAS
1	30	150–200	25	0.417	0.062	0.646	0.428	0.080	0.688	0.495	0.086	0.796
2	30	150–200	50	0.641	0.029	0.896	0.440	0.068	0.650	0.514	0.077	0.882
3	30	150–200	75	0.358	0.087	0.453	0.437	0.070	0.615	0.557	0.076	0.645
4	30	150–200	100	0.452	0.084	0.674	0.458	0.088	0.687	0.424	0.144	0.706
5	30	150–200	125	0.458	0.065	0.669	0.537	0.208	0.463	0.587	0.115	1.011
6	30	300–400	25	0.593	0.042	0.872	0.385	0.067	0.508	0.633	0.125	1.095
7	30	300–400	50	0.626	0.041	0.933	0.390	0.073	0.524	0.631	0.124	1.106
8	30	300–400	75	0.482	0.100	0.775	0.460	0.079	0.683	0.621	0.132	0.965
9	30	300–400	100	0.618	0.039	0.920	0.620	0.035	1.032	0.625	0.148	1.069
10	30	300–400	125	0.472	0.103	0.804	0.380	0.074	0.532	0.625	0.148	0.983
11	40	150–200	25	0.390	0.091	0.573	0.317	0.079	0.388	0.178	0.033	0.161
12	40	150–200	50	0.421	0.075	0.577	0.423	0.088	0.646	0.360	0.099	0.269
13	40	150–200	75	0.501	0.040	0.658	0.354	0.075	0.487	0.380	0.097	0.544
14	40	150–200	100	0.457	0.044	0.588	0.302	0.089	0.378	0.336	0.118	0.476
15	40	150–200	125	0.419	0.089	0.682	0.290	0.033	0.319	0.446	0.120	0.768
16	40	300–400	25	0.434	0.063	0.720	0.315	0.059	0.382	0.292	0.054	0.322
17	40	300–400	50	0.418	0.055	0.536	0.337	0.077	0.423	0.449	0.058	0.615
18	40	300–400	75	0.464	0.120	0.830	0.324	0.081	0.437	0.583	0.045	0.848
19	40	300–400	100	0.472	0.092	0.739	0.322	0.082	0.441	0.450	0.059	0.626
20	40	300–400	125	0.434	0.107	0.634	0.313	0.078	0.437	0.453	0.043	0.469
21	50	150–200	25	0.308	0.115	0.433	0.262	0.077	0.304	0.387	0.099	0.379
22	50	150–200	50	0.521	0.077	0.873	0.227	0.075	0.222	0.528	0.143	1.115
23	50	150–200	75	0.352	0.261	0.731	0.194	0.103	0.209	0.461	0.141	0.851
24	50	150–200	100	0.380	0.051	0.437	0.275	0.300	0.508	0.479	0.113	0.731
25	50	150–200	125	0.492	0.154	0.899	0.143	0.021	0.140	0.433	0.081	0.550
26	50	300–400	25	0.261	0.064	0.328	0.271	0.048	0.288	0.407	0.074	0.507
27	50	300–400	50	0.486	0.133	0.949	0.362	0.047	0.451	0.395	0.031	0.479
28	50	300–400	75	0.521	0.252	1.141	0.361	0.129	0.512	0.368	0.041	0.476
29	50	300–400	100	0.673	0.103	1.607	0.343	0.064	0.451	0.415	0.034	0.506
30	50	300–400	125	0.468	0.183	1.099	0.318	0.058	0.348	0.447	0.101	0.631

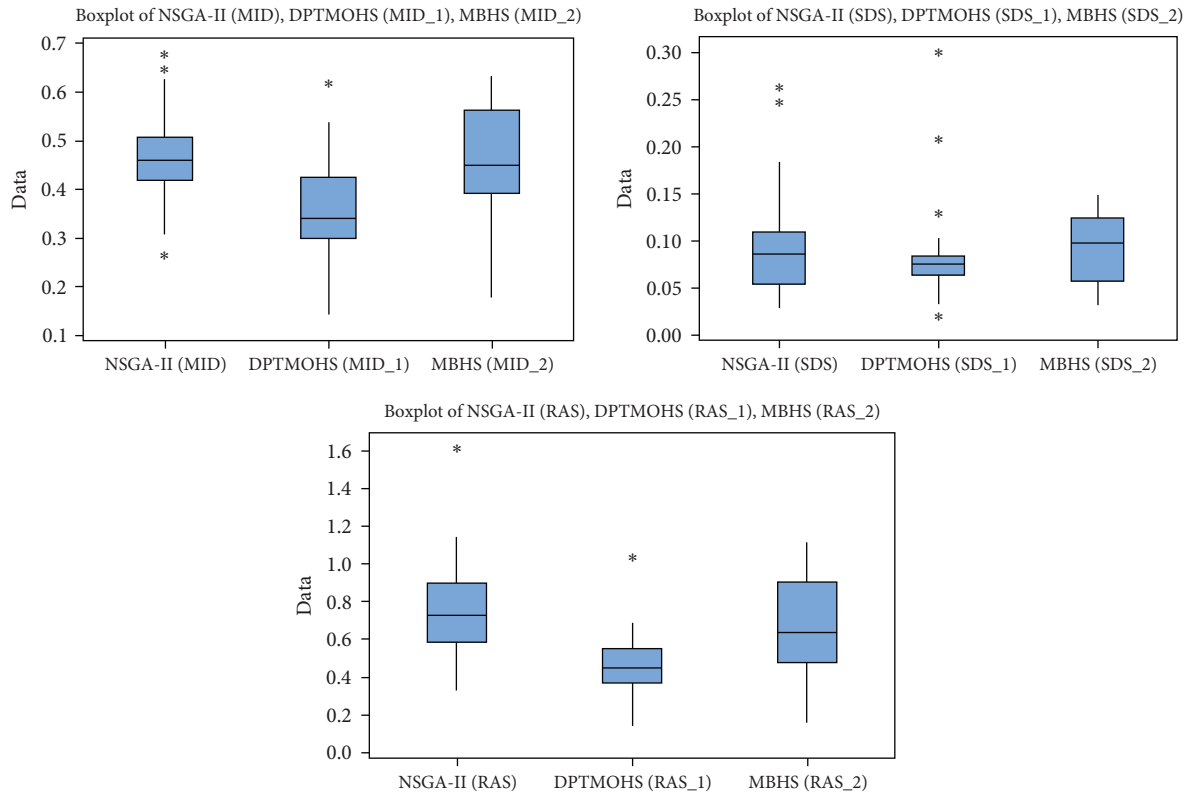


Figure 4: Boxplot of the RAS, MID, SDS indicators (Kruskal–Wallis test) for three algorithm.

Table 13: Paired t-test results to compare DPTMOHS and NSGA-II.

Indicator	Test results					Interpretation
	Paired T for MID − MID_1					
		N	Mean	StDev	SE mean	
MID	MID	30	0.4664	0.0952	0.0174	Null hypothesis is rejected
	MID_1	30	0.3528	0.0977	0.0178	
	Difference	30	0.1135	0.1082	0.0197	
	95% CI for mean difference: (0.0731, 0.1539)					
	T-test of mean difference = 0 (vs. ≠ 0): T-value = 5.75 P-value = 0.00					
	Paired T for SDS − SDS_1					
		N	Mean	StDev	SE mean	
SDS	SDS	30	0.0941	0.0568	0.0104	Null hypothesis is not rejected
	SDS_1	30	0.0835	0.0522	0.0095	
	Difference	30	0.0106	0.0782	0.0143	
	95% CI for mean difference: (−1.0186, 0.0398)					
	T-test of mean difference = 0 (vs. ≠ 0): T-value = 0.74 P-value = 0.463					
	Paired T for RAS − RAS_1					
		N	Mean	StDev	SE mean	
RAS	RAS	30	0.7558	0.2520	0.0460	Null hypothesis is rejected
	RAS_1	30	0.4717	0.1797	0.0325	
	Difference	30	0.1135	0.1082	0.0197	
	95% CI for mean difference: (0.1724, 0.3958)					
	T-test of mean difference = 0 (vs. ≠ 0): T-value = 5.2 P-value = 0.00					

Table 14: Comparison of right-tail and utility function approach.

Test problems	Number of variables	MNPV	E	B	Right tail probability	Utility function
1	40	25	150	200	52.69	66.03
2	40	25	300	400	58,54	96.60
3	40	50	150	200	59.02	57.34
4	40	50	300	400	63.92	85.25
8	40	100	300	400	99.82	94.30
9	40	125	150	200	53.08	81.31
10	40	125	300	400	53.63	95.56
11	50	25	150	200	54,09	79.48
12	50	25	300	400	73.01	73.18
13	50	50	150	200	69.31	79.71
14	50	50	300	400	55.88	108.34
15	50	75	150	200	69.85	80.43
16	50	75	300	400	83.99	148.61
17	50	100	150	200	55.00	74.17
18	50	100	300	400	87.21	133.07
19	50	125	150	200	63.90	74.38
20	50	125	300	400	39.65	65.44
Mean					**64.35**	**87.42**

in the SDS index (P-value = 0.321). However, the RAS (P-value = 0.0001) and MID (P-value = 0.0002) indexes show distinctions. According to the results of the paired t-test, the MID of the DPTMOHS algorithm is superior to that of NSGA-II, while the RAS of NSGA-II surpasses that of the DPTMOHS algorithm.

The proposed approach can be beneficial for VCs aiming to choose investees with potentially higher returns, given certain probability conditions [32]. Furthermore, we introduce a utility function that is more robust in the right tail of the PDNPV approach for portfolio selection optimization. All these methods prove highly useful for VC decision-making and for private equity investors.

6. Conclusion

VCs employ various quantitative and qualitative models for deal valuation and portfolio selection. This study introduces a novel framework for valuing and selecting deals, incorporating a combination of contractual terms, such as call options, liquidity preferences, and participant rights, within VC portfolios. VCs aim to select investee portfolios that not only yield the highest possible returns but also align with the strategic goals of their portfolio. Thus, we have proposed a multiobjective mathematical model that considers both financial and nonfinancial dimensions in VC portfolio selection. In this context, we utilize a numerical integrated method where the NPV of each deal is stochastic, determining the probability of the right tail side of the stochastic NPV portfolio. Given the NP-hard computational complexity of the model, we employ the DPTMOHS algorithm—a metaheuristic based on harmony search—to address and solve the proposed multiobjective model, followed by a thorough assessment of the algorithm's performance. Furthermore, we introduce a new robust utility function tailored for VC firms. This function demonstrates superior resilience compared to the right tail side of the NPV probability function approach when dealing with default. The proposed function empowers VC decision-makers to make more informed and resilient decisions in the face of potential failures. We encourage future researchers to explore exact algorithms to discover the best and optimum Pareto solutions. Furthermore, the integration of various types of contractual terms would significantly contribute to the advancement of this field. Moreover, we propose the development of bi-level mathematical models for deal selection, incorporating diversification indices across various investment stages and geographical locations.

Appendix

$$
\begin{aligned}
\Phi(\mathrm{MNPV}) &= Pr\{S_N > \mathrm{MNPV}\} = Pr\left\{\left(\sum_{i=1}^{N} \mathrm{npv}_i\right) > \mathrm{MNPV}\right\}, \\
&= Pr\{\mathrm{npv}_1 > \mathrm{MNPV}\} + Pr\{\mathrm{npv}_1 + \mathrm{npv}_2 > \mathrm{MNPV} \cap \mathrm{npv}_1 < \mathrm{MNPV}\} \\
&\quad + Pr\{\mathrm{npv}_1 + \mathrm{npv}_2 + \mathrm{npv}_3 > \mathrm{MNPV} \cap \mathrm{npv}_1 + \mathrm{npv}_2 < \mathrm{MNPV}\} \\
&\quad + \cdots + Pr\{\mathrm{npv}_1 + \mathrm{npv}_2 + \cdots + \mathrm{npv}_N > \mathrm{MNPV} \cap \mathrm{npv}_1 + \mathrm{npv}_2 + \cdots + \mathrm{npv}_{N-1} < \mathrm{MNPV}\} \\
&\quad \Phi(\mathrm{MNPV}) = \left\{1 - F_{\mathrm{npv}_1}(\mathrm{MNPV})\right\} \\
&\quad + \int_{-\infty}^{\infty} Pr\{\mathrm{npv}_2 > \mathrm{MNPV} - \mathrm{npv}_1 \cap \mathrm{npv}_1 < \mathrm{MNPV} | \mathrm{npv}_1 = h\} f_{\mathrm{npv}_1}(h)\, dh \\
&\quad + \int_{-\infty}^{\infty} Pr\{\mathrm{npv}_3 > \mathrm{MNPV} - S_2 \cap S_2 < \mathrm{MNPV} | S_2 = h\} f_{S_2}(h)\, dh + \cdots \\
&\quad + \int_{-\infty}^{\infty} Pr\{\mathrm{npv}_{N-1} > \mathrm{MNPV} - S_{N-1} \cap S_{N-1} < \mathrm{MNPV} | S_{N-1} = h\} f_{S_{N-1}}(h)\, dh \\
&\quad \Phi(\mathrm{MNPV}) = \left\{1 - F_{\mathrm{npv}_1}(\mathrm{MNPV})\right\} + \int_{-\infty}^{\mathrm{MNPV}} Pr\{\mathrm{npv}_2 > \mathrm{MNPV} - h\} f_{\mathrm{npv}_1}(h)\, dh \\
&\quad \int_{-\infty}^{\mathrm{MNPV}} Pr\{\mathrm{npv}_3 > \mathrm{MNPV} - h\} f_{\mathrm{npv}_1 + \mathrm{npv}_2}(h)\, dh + \cdots + \int_{-\infty}^{TNPV} Pr\{\mathrm{npv}_N > \mathrm{MNPV} - h\} f_{\mathrm{npv}_1 + \cdots + \mathrm{npv}_{N-1}}(h)\, dh. \\
&\quad \Phi(\mathrm{MNPV}) = \left\{1 - F_{\mathrm{npv}_1}(\mathrm{MNPV})\right\} + \int_{-\infty}^{\mathrm{MNPV}} \left\{1 - F_{\mathrm{npv}_1}(\mathrm{MNPV} - h)\right\} f_{\mathrm{npv}_1}(h)\, dh + \int_{-\infty}^{\mathrm{MNPV}} \left\{1 - F_{\mathrm{npv}_2}(\mathrm{MNPV} - h)\right\} f_{S_2}(h)\, dh \\
&\quad + \cdots + \int_{-\infty}^{TNPV} \left\{1 - F_{\mathrm{npv}_{N-1}}(\mathrm{MNPV} - h)\right\} f_{S_{N-1}}(h)\, dh \\
&= \left\{1 - F_{\mathrm{npv}_1}(\mathrm{MNPV})\right\} + \sum_{i=1}^{N-1} \int_{-\infty}^{\mathrm{MNPV}} \left\{1 - F_{\mathrm{npv}_{N-1}}(\mathrm{MNPV} - h)\right\} f^i(h)\, dh \ , \quad f^i(h) = f_{S_i}(h)
\end{aligned}
\tag{A.1}
$$

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Disclosure

This paper is part of one of the coauthor's PhD thesis in the Industrial Engineering Department at Bu-Ali-Sina University [71].

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Mohammadreza Valaei has contributed to the conceptualization, methodology, software, validation, formal analysis, investigation, data curation, writing—original draft, and Vahid Khodakarami has contributed to the writing—review & editing, supervision, project administration, funding acquisition.

Acknowledgments

This research was funded by the Iran National Science Foundation under grant number (98017405).

References

[1] Y. Pu and S. Fang, "The optimal portfolio size of venture capital under staged financing," *Procedia Computer Science*, vol. 91, pp. 85–93, 2016.

[2] A. Knill, "Should venture capitalists put all their eggs in one basket? Diversification versus pure-play strategies in venture capital," *Financial Management*, vol. 38, no. 3, pp. 441–486, 2009.

[3] D. Klonowski, *The Venture Capital Investment Process*, Springer, 2010.

[4] E. Afful-Dadzie and A. Afful-Dadzie, "A decision making model for selecting start-up businesses in a government venture capital scheme," *Management Decision*, vol. 54, no. 3, pp. 714–734, 2016.

[5] X. Zhang, "Venture capital investment selection decision-making base on fuzzy theory," *Physics Procedia*, vol. 25, pp. 1369–1375, 2012.

[6] D. Dimov and D. De Clercq, "Venture capital investment strategy and portfolio failure rate: a longitudinal study," *Entrepreneurship Theory and Practice*, vol. 30, no. 2, pp. 207–223, 2006.

[7] B. Elango, V. H. Fried, R. D. Hisrich, and A. Polonchek, "How venture capital firms differ," *Journal of Business Venturing*, vol. 10, no. 2, pp. 157–179, 1995.

[8] D. Cumming, *Venture Capital: Investment Strategies, Structures, and Policies*, John Wiley & Sons, 2010.

[9] P. A. Gompers, W. Gornall, S. N. Kaplan, and I. A. Strebulaev, "How do venture capitalists make decisions?" *Journal of Financial Economics*, vol. 135, no. 1, pp. 169–190, 2020.

[10] A. Weissmann, "Impact of real options on venture valuations," 2015.

[11] W. Gornall and I. A. Strebulaev, "A valuation model of venture capital-backed companies with multiple financing rounds," Available at SSRN 3725240, 2021.

[12] B. Payne, "Scorecard valuation methodology," 2010 Angel Valuation Survey, 2011.

[13] G. Behrmann, *Internet Company Valuation: A Study of Valuation Methods and Their Accuracy*, EBS Universität für Wirtschaft und Recht, Hamburg, 2016.

[14] L. Yan, Z. Hong, and H. Lucheng, "Review on methods of new technology valuation," in *2010 International Conference on*

E-Business and E-Government, pp. 1932–1935, IEEE, Guangzhou, China, 2010.

[15] K. Guo, L. Zhang, and T. Wang, "Optimal scheme in energy performance contracting under uncertainty: a real option perspective," *Journal of Cleaner Production*, vol. 231, pp. 240–253, 2019.

[16] S. Li, D. Hu, J. Cai, and H. Cai, "Real option-based optimization for financial incentive allocation in infrastructure projects under public-private partnerships," *Frontiers of Engineering Management*, vol. 7, pp. 413–425, 2020.

[17] M. Montajabiha, A. A. Khamseh, and B. Afshar-Nadjafi, "A robust algorithm for project portfolio selection problem using real options valuation," *International Journal of Managing Projects in Business*, vol. 10, no. 2, pp. 386–403, 2017.

[18] G. L. Nigro, G. Enea, and A. Morreale, "A user friendly real option based model to optimize pharmaceutical R&D portfolio," *International Journal of Applied Operational Research*, vol. 5, pp. 83–95, 2013.

[19] J. D. MacBeth and L. J. Merville, "An empirical examination of the Black-Scholes call option pricing model," *The Journal of Finance*, vol. 34, no. 5, pp. 1173–1186, 1979.

[20] P. Morales-Bañuelos, N. Muriel, and G. Fernández-Anaya, "A modified Black-Scholes-Merton model for option pricing," *Mathematics*, vol. 10, no. 9, Article ID 1492, 2022.

[21] S.-P. Zhu and X.-J. He, "A modified Black–Scholes pricing formula for European options with bounded underlying prices," *Computers & Mathematics with Applications*, vol. 75, no. 5, pp. 1635–1647, 2018.

[22] M. Barbati, S. Corrente, and S. Greco, "A general space-time model for combinatorial optimization problems (and not only)," *Omega*, vol. 96, Article ID 102067, 2020.

[23] X. A. F. Chaparro, L. A. de Vasconcelos Gomes, and P. T. de Souza Nascimento, "The evolution of project portfolio selection methods: from incremental to radical innovation," *Revista de Gestão*, vol. 26, no. 3, pp. 212–236, 2019.

[24] Y. Guo, L. Wang, S. Li, Z. Chen, and Y. Cheng, "Balancing strategic contributions and financial returns: a project portfolio selection model under uncertainty," *Soft Computing*, vol. 22, no. 16, pp. 5547–5559, 2018.

[25] B. Aouni, C. Colapinto, and D. La Torre, "A cardinality constrained stochastic goal programming model with satisfaction functions for venture capital investment decision making," *Annals of Operations Research*, vol. 205, no. 1, pp. 77–88, 2013.

[26] B. Aouni, C. Colapinto, and D. La Torre, "A fuzzy goal programming model for venture capital investment decision making," *INFOR: Information Systems and Operational Research*, vol. 52, no. 3, pp. 138–146, 2014.

[27] X. Zhao, W. Zhang, and J. Wang, "Risk-hedged venture capital investment recommendation," in *Proceedings of the 9th ACM Conference on Recommender Systems*, pp. 75–82, ACM, 2015.

[28] D. J. Cumming, "The determinants of venture capital portfolio size: empirical evidence," *The Journal of Business*, vol. 79, no. 3, pp. 1083–1126, 2006.

[29] M. Nitani and A. Riding, "Fund size and the syndication of venture capital investments," *Venture Capital*, vol. 15, no. 1, pp. 53–75, 2013.

[30] S. Manigart, K. De Waele, M. Wright et al., "Determinants of required return in venture capital investments: a five-country study," *Journal of Business Venturing*, vol. 17, no. 4, pp. 291–312, 2002.

[31] B. Huntsman and J. P. Hoban Jr., "Investment in new enterprise: some empirical observations on risk, return, and market structure," *Financial Management*, vol. 9, no. 2, pp. 44–51, 1980.

[32] A. Retterath and S. Kavadias, "How to hit home runs: portfolio strategies and returns in formal and informal venture capital," Available at SSRN 3527412, 2020.

[33] S. de Treville, J. S. Petty, and S. Wager, "Economies of extremes: lessons from venture-capital decision making," *Journal of Operations Management*, vol. 32, no. 6, pp. 387–398, 2014.

[34] J. Charnes, *Financial Modeling with Crystal Ball and Excel,+ Website*, John Wiley & Sons, 2012.

[35] N. Larsen, H. Mausser, and S. Uryasev, *Algorithms for Optimization of Value-at-Risk*, Springer, 2002.

[36] F. Hooshmand, Z. Anoushirvani, and S. A. MirHassani, "Model and efficient algorithm for the portfolio selection problem with real-world constraints under value-at-risk measure," *International Transactions in Operational Research*, vol. 30, no. 5, pp. 2665–2690, 2023.

[37] A. H. Ayob and L.-P. Dana, "Product strategies for export ventures: an empirical investigation among SMEs in an emerging economy," *Jurnal Pengurusan (UKM Journal of Management)*, vol. 50, pp. 25–33, 2017.

[38] S. Park and Z.-T. Bae, "New venture strategies in a developing country: identifying a typology and examining growth patterns through case studies," *Journal of Business Venturing*, vol. 19, no. 1, pp. 81–105, 2004.

[39] H. Zhong, C. Liu, J. Zhong, and H. Xiong, "Which startup to invest in: a personalized portfolio strategy," *Annals of Operations Research*, vol. 263, no. 1-2, pp. 339–360, 2018.

[40] T. A. Herberger and F. Reinle, "A framework for screening and portfolio selection in corporate venture capital," *Studies in Economics and Finance*, vol. 37, no. 3, pp. 475–495, 2020.

[41] S. A. Hasheminejad, M. Bagherpour, S. Nouri, and M. S. Pishvaee, "Syndicated venture capital portfolio companies selection: a fuzzy inference system—agent-based approach," *International Journal of Computer Mathematics*, vol. 98, no. 11, pp. 2186–2201, 2021.

[42] M. Valaei and V. Khodakarami, "A new multi-dimensional framework for start-ups lifespan assessment using Bayesian networks," *Journal of Risk and Financial Management*, vol. 16, no. 2, Article ID 88, 2023.

[43] F. Black and M. Scholes, "The pricing of options and corporate liabilities," *Journal of Political Economy*, vol. 81, no. 3, pp. 637–654, 1973.

[44] K. Reddy and V. Clinton, "Simulating stock prices using geometric brownian motion: evidence from Australian companies," *Australasian Accounting, Business and Finance Journal*, vol. 10, no. 3, pp. 23–47, 2016.

[45] M. Hifi, M. Michrafy, and A. Sbihi, "Heuristic algorithms for the multiple-choice multidimensional knapsack problem," *Journal of the Operational Research Society*, vol. 55, no. 12, pp. 1323–1332, 2004.

[46] S. Benati and R. Rizzi, "A mixed integer linear programming formulation of the optimal mean/value-at-risk portfolio problem," *European Journal of Operational Research*, vol. 176, no. 1, pp. 423–434, 2007.

[47] K. Hans, P. Ulrich, and P. David, *Knapsack Problems*, Spinger-Verlag, Berlin, 2004.

[48] X. Kong, L. Gao, H. Ouyang, and S. Li, "Solving large-scale multidimensional knapsack problems with a new binary harmony search algorithm," *Computers & Operations Research*, vol. 63, pp. 7–22, 2015.

[49] V. Cacchiani, M. Iori, A. Locatelli, and S. Martello, "Knapsack problems—an overview of recent advances. Part II: multiple,

multidimensional, and quadratic knapsack problems," *Computers & Operations Research*, vol. 143, Article ID 105693, 2022.

[50] P. B. Soares, M. R. N. Guimarães, and F. F. de Lara, "Analysis of the internal strategic alignment of the production strategy: case study in an auto parts company of the municipality of Sorocaba/SP," *Gestão & Produção*, vol. 26, no. 4, Article ID e2762, 2019.

[51] G. J. C. da Silveira, "Market priorities, manufacturing configuration, and business performance: an empirical analysis of the order-winners framework," *Journal of Operations Management*, vol. 23, no. 6, pp. 662–675, 2005.

[52] A. Buchner, A. Mohamed, and A. Schwienbacher, "Diversification, risk, and returns in venture capital," *Journal of Business Venturing*, vol. 32, no. 5, pp. 519–535, 2017.

[53] R. Cressy, A. Malipiero, and F. Munari, "Does VC fund diversification pay off? An empirical investigation of the effects of VC portfolio diversification on fund performance," *International Entrepreneurship and Management Journal*, vol. 10, no. 1, pp. 139–163, 2014.

[54] D. W. Coit, "Cold-standby redundancy optimization for nonrepairable systems," *IIE Transactions*, vol. 33, no. 6, pp. 471–478, 2001.

[55] M. A. Akbay, C. B. Kalayci, and O. Polat, "A parallel variable neighborhood search algorithm with quadratic programming for cardinality constrained portfolio optimization," *Knowledge-Based Systems*, vol. 198, Article ID 105944, 2020.

[56] S. Hong, K. Serfes, and V. Thiele, "Competition in the venture capital market and the success of startup companies: theory and evidence," *Journal of Economics & Management Strategy*, vol. 29, no. 4, pp. 741–791, 2020.

[57] R.-J. Luo, S.-F. Ji, and B.-L. Zhu, "A Pareto evolutionary algorithm based on incremental learning for a kind of multi-objective multidimensional knapsack problem," *Computers & Industrial Engineering*, vol. 135, pp. 537–559, 2019.

[58] X. Kong, L. Gao, H. Ouyang, and S. Li, "A simplified binary harmony search algorithm for large scale 0–1 knapsack problems," *Expert Systems with Applications*, vol. 42, no. 12, pp. 5337–5355, 2015.

[59] D. Zou, L. Gao, S. Li, and J. Wu, "Solving 0–1 knapsack problem by a novel global harmony search algorithm," *Applied Soft Computing*, vol. 11, no. 2, pp. 1556–1564, 2011.

[60] M. R. Valaei and J. Behnamian, "Allocation and sequencing in 1-out-of-N heterogeneous cold-standby systems: multi-objective harmony search with dynamic parameters tuning," *Reliability Engineering & System Safety*, vol. 157, pp. 78–86, 2017.

[61] M. Valaei, "A new meta-heuristic algorithm based on harmony search to optimize the reliability of parallel-systems by considering the allocation of redundant series components," in *Industrial Engineering*, A. Kheirkhah, Ed., Bual Sina University, Hemedan, 2015.

[62] Z. W. Geem, J. H. Kim, and G. V. Loganathan, "A new heuristic optimization algorithm: harmony search," *SIMULATION*, vol. 76, no. 2, pp. 60–68, 2001.

[63] N. Nahas and D. Thien-My, "Harmony search algorithm: application to the redundancy optimization problem," *Engineering Optimization*, vol. 42, no. 9, pp. 845–861, 2010.

[64] M. Mahdavi, M. Fesanghary, and E. Damangir, "An improved harmony search algorithm for solving optimization problems," *Applied Mathematics and Computation*, vol. 188, no. 2, pp. 1567–1579, 2007.

[65] L. Wang, L. An, H.-Q. Ni, W. Ye, P. M. Pardalos, and M.-R. Fei, "Pareto-based multi-objective node placement of industrial wireless sensor networks using binary differential evolution harmony search," *Advances in Manufacturing*, vol. 4, no. 1, pp. 66–78, 2016.

[66] L. Wang, Y. Mao, Q. Niu, and M. Fei, "A multi-objective binary harmony search algorithm," in *Advances in Swarm Intelligence. ICSI 2011*, vol. 6729 of *Lecture Notes in Computer Science*, pp. 74–81, Springer, Berlin Heidelberg, 2011.

[67] J. Behnamian and Z. Gharabaghli, "Multi-objective outpatient scheduling in health centers considering resource constraints and service quality: a robust optimization approach," *Journal of Combinatorial Optimization*, vol. 45, no. 2, Article ID 80, 2023.

[68] D. Abbasi, M. Ashrafi, and S. H. Ghodsypour, "A multi objective-BSC model for new product development project portfolio selection," *Expert Systems with Applications*, vol. 162, Article ID 113757, 2020.

[69] X. Li, B. Li, T. Jin, and P. Zheng, "Uncertain random portfolio optimization with non-dominated sorting genetic algorithm-II and optimal solution criterion," *Artificial Intelligence Review*, vol. 56, no. 8, pp. 8511–8546, 2023.

[70] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[71] M. Valaei, *Valuation and optimization of venture capital deals portfolio based on risk approach*, PhD thesis, Industerial engineering Department https://eng.basu.ac.ir/en/ie-seminars, Bu-Ali Sina University, Hamedan, Iran, 2023.

Hindawi
Mathematical Problems in Engineering
Volume 2023, Article ID 9937949, 9 pages
https://doi.org/10.1155/2023/9937949

Research Article

An Energy Efficient Evolutionary Approach for Smart City-Based IoT Applications

Rashmi Prava Das,[1] Tushar Kanta Samal,[1] and Ashish Kr. Luhach[2]

[1]*Department of Computer Science and Engineering, CV Raman Global University, Bhubaneswar, India*
[2]*The Papua New Guinea University of Technology, Lae, Papua New Guinea*

Correspondence should be addressed to Ashish Kr. Luhach; ashish.kumar@pnguot.ac.pg

Received 20 February 2023; Revised 6 November 2023; Accepted 11 November 2023; Published 7 December 2023

Academic Editor: Anna M. Gil-Lafuente

Internet of Things (IoT) has been used in smart cities, agriculture, weather forecasting, smart grids, and waste management. The IoT has huge potential but needs refinement. The paper focuses on lowering IoT sensor power consumption to improve network life. This work selects the best IoT cluster header (CH) to maximize energy consumption. The suggested technique uses particle swarm optimization (PSO) with artificial neural networks (ANNs). The optimal CH in an IoT network cluster was identified by taking into account the number of active nodes, the load, the residual energy, and the cost function. This work compares the suggested method with artificial bee colony, genetic, and adaptive gravity search algorithms. The hybrid solution beats conventional methods.

1. Introduction

A smart city is an urban environment that uses information and communication technology (ICT) and other closely related technologies to improve the efficiency of everyday urban activities and the quality of service (QoS) provided to urban residents [1, 2]. Continued population growth and urbanization are driving the need for creative approaches to urban management with minimal impact on the environment, public life, and governance. The process of integrating intelligent urbanization has been facilitated through the use of ICT in city operations. With the proliferation of smart devices and recent advances, the idea of smart cities has gained a lot of attention. Internet of Things (IoT) has been used in smart cities, agriculture, weather forecasting, smart grids, and waste management etc [3, 4]. As a result of the Internet of Things (IoT), traditional networks have emerged, connecting millions of connected devices. Further, realization of this IoT concept in smart cities requires continuous technological advances in ubiquitous computing, wireless sensor networks (WSNs), machine-to-machine communication, and design of cross-industry applications.

The de facto policy of the IoT is to enable unified communications through uniquely identifiable smart devices with minimal human intervention. The IoT paradigm has attracted significant interest from a wide variety of interest groups, including smart city, smart home, smart health, and many others.

Getting everything from the cloud is getting more complicated. Cloud computing provides a variety of resources, such as storage and processing, that you can access when you need them. Despite its many advantages, there are also some drawbacks that prevent full adoption of the cloud and realization of its various advantages. The problems we faced with cloud infrastructure adaptability were high latency, high power consumption, and uncertain geographic location.

To address these issues, the idea of fog computing was developed to improve and ensure real-time responsiveness while minimizing cloud load [5]. End users and cloud data centers are usually separated by a layer of fog. Edge detection, location detection, geographical edge adaptation, wireless network access, and power reduction are the main functions of this core. In most cases, it is not practical to exceed the amount of power consumed by a cloud data center. Some energy consumption can be minimized, but the fog layer uses small self-powered data centers to make minor adjustments to minimize power consumption. Fog computing can help manage smart devices by adding data to the network's edge. Fog is context aware, low latency, and mobile. As IoT devices

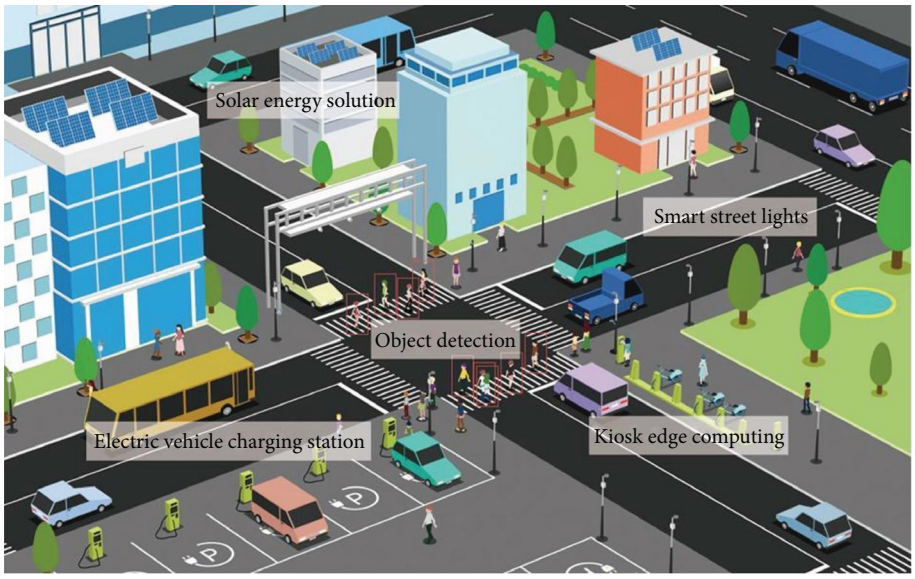


Figure 1: A complex smart city scenario with ICT infrastructure conceptual layers.

proliferate, cloud utilization increases. Due to fog's popularity, researchers monitor energy use. Figure 1 shows the detailed layout of the access point and fog gateway, and section Abbreviations describe the abbreviation used in this research work. Using context-aware computing at the 45-fog layer has been proven successful in processing data streams from various IoT applications [6–8].

The fog layer computings are intended to offload tasks from the cloud layer and increase service response time [9, 10]. As sustainability is one of the most pressing issues in smart infrastructure today, energy conservation at all levels of smart city IoT infrastructure is an important research work [11, 12].

Numerous research projects have been conducted on the topic of energy-saving strategies that can be found in the device or sensor layer [13–15]. Duty cycle mechanisms play an important role in reducing energy consumption at both system and network levels in IoT environments [16–19]. Dhall and Agrawal [16] introduced an improved duty cycling algorithm that uses effective path selection methods based on residual energy parameters to minimize power consumption and maximize network reliability and lifetime. Amirtharaj et al. [17] proposed an algorithm to improve the duty cycle efficiency of Linux-based IoT devices by finding and deactivating units that take up time and aren't necessary that are initialized in user space. It presents an energy-efficient context-aware traffic scheduling method that first defines IoT applications based on the diverse traffic requirements they have and then assigns those apps to various weighted quality classes [18]. Kozłowski and Sosnowski [19] proposed various power consumption models for the various duty cycle schemes and the maximum power level consumed by the additional wake-up system was determined to determine commonly used duty cycles in typical IoT networks that made it a fair alternative to the cycle method. IoT and WSN environments typically consist of large numbers of low-cost, battery-powered devices with limited functionality. Because IoT devices are frequently installed in hospitable and unmanned environments, it is impractical to change their batteries. Therefore, the most critical aspect in the process of creating an effective IoT or WSN is the conservation of power [7, 8, 20–22].

Early WSN network research focused on smart grids and environmental and agricultural monitoring, utilizing low-rate, delay-tolerant techniques. Smart cities, industrial automation, healthcare, transit automation, and multimedia use IoT technologies [7, 8]. These are classified as delay-sensitive applications and must be considered for timing constraints. Therefore, the second most important issue in IoT is node-to-node and end-to-end latency. Furthermore, it is difficult to ensure a given delay [20–22]. In different areas of IoT and WSN, researchers have proposed different energy harvesting approaches [21, 23–25] to meet the demand of different energy sources. The models proposed by Barath et al. [26], Zhan and Lai [27], and Sendra Compte et al. [28] are useful for predicting potential energy demand based on historical evidence. On the other hand, if the harvest is uncontrolled or unpredictable, this algorithm will not work well. Various models have been proposed to overcome this limitation by Beheshtiha et al. [29], Briante et al. [30], Oliveira and Castro [31], and Oliveira et al. [32], with the aim of balancing power supplies and minimizing dispersion at different sensor nodes under different environmental conditions. Various update algorithms are used. A number of references [33, 34] often deal with energy minimization in cloud layers.

Fog devices are commonly employed in IoT contexts, and study shows fog node middleware has more computational capacity than end devices but less than cloud centers. Ma et al. [35] proposed this IoT-based fog computing model that used genetic algorithms to reduce faulty nodes and power consumption. IoT networks are efficiently handled

using fog nodes, edge nodes, or a combination of cloud and fog by Skarlat et al. [36], Souza et al. [37], Minh et al. [38], Naas et al. [39], Taneja and Davy [40], and Lera et al. [41].

Various service deployment methods have been proposed for different platforms to reduce power consumption. Similarly, there are many publications where authors used different optimization strategies to reduce energy consumption at fog and edge levels. In contrast, advances in fog layer research in smart city research have only recently been recognized for their relevance and research potential [42–44]. Examples of such applications of scenarios include large-scale video summarization for smart cities [42], urban traffic monitoring [43], and automatic adjustment of surveillance cameras for smart cities [44]. However, little research has focused on the energy savings of fog layers. To further enhance the effectiveness of performance mediation, some authors apply different optimization strategies to optimize performance mediation at different levels [45–49]. An optimized model based on genetic algorithms, proposed in a study by Skarlat et al. [45], considers application and resource QoS heterogeneity to install this IoT application using fog resources. Zubair et al. [46] and Sun et al. [47] used a multiobjective optimization technique to solve the placement of services on different platforms. Guerrero et al. [48] gave a comparative overview of different optimization techniques. Kaur and Sood [49] presented forecasting and predictive models that solve work scheduling problems using artificial neural networks and genetic algorithms.

Traffic, load, and temperature make IoT increasingly energy intensive. Therefore, a high-energy-efficiency IoT architecture must be proposed to ensure network stability and lifespan. For this purpose, this study uses a hybrid particle swarm optimization-neural network (PSO-NN) to find the optimal cluster header (CH) in specific IoT networks. This hybrid strategy begins with a search for the most significant regions that have been uncovered by the PSO algorithm. Next, the NN is used to increase utilization of the found regions.

Listed below are some of the primary contributions made by the hybrid model that was suggested.

(1) Select the optimal CH using the PSO hybrid model.
(2) Utilize the PSO-NN model in order to improve the performance metrics of WSN sensors, namely those that have an impact on the amount of energy consumed, such as load, temperature, delay, and distance between BT and CH.

The rest is structured like this: The most recent research on how to optimize the power consumption of WSN sensors in IoT systems is discussed in detail in Section 2. Section 3 gives architectural context. Section 4 outlines the design proposal. Section 5 analyzes experimental data. Section 6 discusses conclusions and future study.

2. Related Works

IoT networks have been expanded in several ways. This section examines renowned works. Neuro-fuzzy rule clustering improves end-to-end efficiency and packet delivery rate. This technique helps forward packets effectively and extends network life [11]. Calculating CH's leftover energy and distance to the sink node offer network efficiency. Simulations suggest that neuro-fuzzy reduces energy consumption and increases network longevity. The creators of this work believe that all network nodes can be trusted; however, this isn't necessarily true.

A methodology for choosing the best cluster header (CH) is also suggested. Short battery life, memory, and communication distances characterize sensor nodes. This communication protocol conveyed the authors' data to IoT BT [12]. Two of the best CHs within the same cluster are selected during the CH selection process to extend the network lifetime. The method of choosing dual CH is data entropy knowledge fusion. Entropy data are used for classification and fusion. Dinesh Reddy et al. [13] combined moth flame optimization (MFO) and ant lion optimization (ALO) to choose the ideal CH in WSN-IoT networks. This methodology selects the best CH by choosing the node nearest to the BT, transmits data quickly, saves node power, and reduces IoT device load and temperature. ABC, AGSA, PSO, GA, ALO, GSA, and MFO were compared to the hybrid model. Modern hybrid models store more WSN-IoT resources. It constructs an energy-aware CH selection model for WSN-IoT networks by using SAWOA as its primary tool. The delay, the amount of time, the power, the load, and the temperature were used to determine CH performance [14].

The proposed model is compared to a number of algorithms based on WOA, including CH, GSA, ABC, GA, AGSA, and PSO selection models. The simulation model's results show that SAWOA is effective in selecting a CH in order to extend network life. Beloglazov and Buyya [15] used a novel MOFGSA algorithm to choose the best CH. The energy of each node in the IoT initially supplies the packets in a significant amount for efficient routing. Fractional theory and GSA are combined in this FGSA algorithm.

The effectiveness of the algorithm is evaluated alongside that of other algorithms already in existence, such as ABC, GSA, multiparticle swarm cooperative algorithm, and so on to ensure a longer IoT node lifespan. Dhall and Agrawal [16] proposed this HEEQA algorithm to achieve a balance between system energies. Then, tune the message authentication code layer parameters to reduce system power consumption. With 190 of these IoT computers, achieving QoS is a big challenge, and maintaining power balance is critical to extending sensor life. Combining quantum PSO with modern uncontrolled gene sorting methods allows for the accomplishment of this goal. The HEEQA algorithm optimizes power consumption and improves network durability, throughput, and coverage according to simulation results. Amirtharaj et al. [17] proposed unequal clustering of time delay routing techniques to overcome WSN power consumption and data transmission problems. It's compared to others. This approach enhances network efficiency and balances energy usage, increasing network life. The proposed technique is ideal for low-latency IoT applications.

A Hy-IoT algorithm is proposed, which extends the performance of clustering to encompass actual cyber-IoT infrastructures [20]. To take advantage of the CH region, weightable election probabilities are updated based on residual strength, distance, and observed heterogeneity conditions by looking at different dynamic steps. Compared to LEACH, SEP, and Z-SEP, simulations show that Hy-IoT extends network life and improves performance. Vigorito et al. [21] proposed a new OGMAD approach that modifies the active period superframe, so that it corresponds to the data that were requested. This approach improves link utilization while guaranteeing more time slot nodes. Hu et al. [22] proposed a context knowledge-based cryptographic protocol for IoT networks that select the optimal cryptographic protocol based on data sensitivity and system requirements.

This approach reduced execution time by 68% while saving 79% on memory usage and 56% on battery usage. Hsu et al. [23] proposed an IoT- and environment-based robot architecture that allows the robot to communicate with any computer on this IoT network. Jurdak et al. [24] used the sensor routing protocol in many applications and also presented a neuro-fuzzy approach for identifying intruders in low-power WSN network. Lee and Chung [25] provided a system to detect and monitor cloud assaults that can be applied to IoT networks. Barath et al. [26] and Zhan and Lai [27] offered a unique IoT-based mechanism to improve home surveillance utilizing smartphone apps and online apps. This strategy protects consumers from home break-ins. From the above, we may conclude that despite the many CH selection models, they all spend a lot of energy. We employ a hybrid PSO-NN algorithm to optimize power consumption by selecting the optimal CH.

3. IoT-Based Adaptive Cluster Head Selection

An IoT network is composed of numerous sensor nodes, each of which has limited memory and requires a significant amount of electricity [24, 28–30]. These nodes are constantly producing data, so the battery is used more often. High-energy consumption shortens network life. One of the energy optimization strategies is choosing the optimal CH. Clustering groups sensor nodes and assigns a leader depending on criteria. A cluster is called a group and a CH is called a group leader.

In the present paper, it considers C nodes in each cluster denoted by Cj, where $J = 1, 2.., N$. Y nodes in each cluster are presented by Yj, where $J = 1, 2, \ldots, M$. TCH stands for total number of CHs. Only the TCH that has been selected is able to interact with the IoT base station. Choosing the best CH for an IoT-based WSN that would maximize uptime has increased challenging.

3.1. Mathematical Formulation of Fitness Function. Traditional WSNs like distance, delay, and power prefer CH. Load and temperature should be considered when combining WSN and IoT. CH was chosen to improve network capacity and endurance by supporting high power nodes with low load, latency, distance, and temperature. Maximizing the fitness function improves network stability and efficiency, which is given in Equation (1):

$$FFun_i = wt0 * FFun_{Temp} + wt1 * FFun_{Load} + wt2 * FFun_{Ener} + wt4 * (1 - FFun_{Dist}) + wt5 * (1 - FFun_{Dela}), \quad (1)$$

where the weighted parameters are wt0, wt1, wt3, wt4, and w5 and the fitness function ($FFun_i$) is the sum of those values. The following subsections illustrate the mathematically modeled calculation of the four parameters outlined during this experiment.

3.2. Energy Computation. In IoT systems, the energy that is absorbed along a single path will be split into two distinct halves at some point. The first component is the total amount of power amplifier energy, denoted by E_{am}; the other component is the total amount of energy used by the other circuit blocks, denoted by E_{cb}. The following is the formula that is used to get the total amount of energy that is consumed by each link:

$$E_c = E_{am} + E_{cb}\ldots. \quad (2)$$

It is not possible to replenish the energy contained in IoT nodes. The initial amount of energy that the IoT node has is denoted by $E_c = 0$. Every node that is part of a cluster is responsible for sending the packets to the CH. Throughout transmittal of packets from xth-specific node to yth CH as nodes lose energy. IoT nodes have receiver and transmitter hardware. A node's energy lost as a transmitter and receiver when it sends or receives data. The transmitter's energy dissipation may be from power electronics or radiophysics, while the receiver's is from radiophysics. Equation (3) defines the node as transferring X bytes of data to CH. Equation (5) demonstrates energy dissipation when CH receives X bytes from a typical node.

The resources of IoT nodes are unable to be replenished. The initial energy of the IoT node is denoted by the value $E_c = 0$. Every node that is part of the cluster is responsible for sending the packet to the CH. Each CH and node lose energy during packet transmission from the xth node to the yth CH. Each IoT node is equipped with receiver and transmitter hardware. Energy is lost as a transmitter and receiver when a node transmits or receives specific data. Energy dissipation in the transmitters could be due to power electronic equipment.

There are two distinct methods for releasing stored energy. The first scenario is depicted in Equation (3) and occurs when the node is being transferred X bytes of data to the CH. Equation (5) describes the energy dissipation that occurs when the CH gets the information of X bytes from the conventional node.

$$E_c(Dist_{N0}^{a0}) = E_{ee} * x) + E_{fes} * x)\,||\,Dist_{N0}^{a0} - Dist_{CluH}^{n0}||, \quad (3)$$

where $E_c(Dist_{N0}^{a0})$ is the energy dissipation of the a0th conventional node, E_{ee} represents electronic energy, and E_{fes} represents free energy house.

and,

$$E_{ee} = E_{te} + E_{dae}, \tag{4}$$

where E_{te} is being transmitter of energy and E_{dae} is information aggregation energy.

$$E_e\left(Dist^n_{CluH}\right) = E_{ee} \text{ *x}. \tag{5}$$

Following the data send and receive operations, the energy levels in all of the conventional nodes and the CH should be altered. Equation (6) gives the modified energy available inside the traditional node to the CH. Equation (7) gives the energy modifications in CH after receiving the information.

Information is sent to CH through the conventional node until the energy of the node becomes nonzero. A dead node has its energy level zero and Equation (8) presents the fitness function related to energy. If the node has high energy, CH should be chosen.

$$E_{E+1}(Dist^{a0}_{N0}) = E_E(Dist^{a0}_{N0}) - E(Dist^{a0}_{N0}). \tag{6}$$

Equation (7) defines the receiving data from the typical node, and it provides the changed energy that is readily available in CH.

$$E_{E+1}\left(Dist^{n0}_{CluH}\right) = E_E\left(Dist^{n0}_{CluH}\right) - E\left(Dist^{n0}_{CluH}\right). \tag{7}$$

The regular node will continue to transfer the data to the CH as long as the energy level of the node is greater than zero. A node is considered to be "dead" whenever its energy level reaches 0, at which point it is removed from the network. Equation (8) is used to represent the fitness function for energy.

$$FitFun_{Ener} = \frac{1}{Y}\left\{\sum_{i=1}^{Y} E_c(Dist^{a0}_{N0})\right\} + \frac{1}{T_{CluH}}\left\{\sum_{n=1}^{CluH} E_e\left(Dist^{n0}_{CluH}\right)\right\}. \tag{8}$$

To be selected as CH, the node's energy must be high.

3.3. Compute Distance. The fitness function of the space between detector nodes and the IoT Bstn is modeled mathematically in Equation (9), which explains how it works. In order to pick the CH that is the least difficult, the distance that separates the nodes from the IoT Bstn should be the shortest possible.

$$FitFun_{Dist} = \sum_{n=1}^{CluH} \frac{\left|\left|Dist^{a0}_{N0} - Dist^{n0}_{CluH}\right|\right| + \left|\left|Dist^{n0}_{CluH} - Dist_{BStn}\right|\right|}{(P * Q)}, \tag{9}$$

where $Dist^{a0}_{N0} - Dist^{n0}_{CluH}$ is the difference between the a_0th traditional node and the $n0$th, $Dist^{n0}_{CluH} - Dist_{BStn}$ is distance between the nth CH and the IoT Bstn, and P and Q (in denominator) are the dimension ranges (in meters).

3.4. Delay Computation. In order to select the most straightforward CH, the delay ought to be as brief as is practicable. It is required that the delay can be between 0 and 1 ms. Because the length of the delay is proportional to the number of nodes in the cluster, you should eliminate nodes from the cluster in order to shorten the length of the delay. Equation (10) provides a mathematical description of the fitness function for delay transmission between devices connected to the IoT and CH. The dividend is a representation of the majority of the information that was transferred from CH to Bstn, while the divisors represent the individual nodes.

$$FitFun_{Dela} = \frac{Max\sum_{n=1}^{T_{CluH}} CluH_{n0}}{Z}. \tag{10}$$

4. PSO-Based Parameter Optimization Model

Kennedy and Eberhart created the PSO in 1995, and it was influenced by the actions of animal groups such as fish, swarms, and bird [23].

The PSO is an iterative optimization approach that is easy to apply, scalable, resilient, and fast to converge. It uses simple mathematical operators and is memory and performance efficient [50]. The PSO method is composed of a horde of particles, each of which stands for a different feasible approach to resolving the issue at hand.

The location, velocity, and fitness value of each particle are all determined by an optimization function. The value of the particle's velocity gives information about both the direction and the distance of its movement. The procedure described in the proposed work begins with the initialization of a group of random particles for L, N, and E. After that, it updates generations in order to search for the best possible answer. Each particle iteration uses the two "best" values. First is the best, thus, far. The particle swarm optimizer tracks each particle's best value so far. gbest is a global best value (the global best position). Each particle updates its location and velocity by monitoring pbest and gbest and the velocity and location of the particles are modified using the below mentioned equations:

$$\begin{aligned} V_{i,j}(t+1) = {} & w * V_{i,j}(t) + c1 * r1 * \left(p^{lB}_{ij}(t) - x_{i,j}(t)\right) \\ & + c2 * r2 * p^{gB}_j(t) - x_{i,j}(t)), \end{aligned} \tag{11}$$

$$x_{i,j}(t+1) = x_{i,j}(t) + v_{i,j}(t+1), \tag{12}$$

where discovery and exploitation abilities of PSOs are impartial by the inertia weight ω. Two distinct random numbers, viz., $r1$ and $r2$ are (0, 1), contributing to the algorithm's existence. The acceleration coefficients ($c1$, $c2$) force each

Initialization of x_i, v_i, c_1, c_2, l_b, u_b
While $I_{current} <= I_{\max}$ **do**
1. Map x_i into **W** and b
2. Evaluate Equations 11 and 12. This phase is called training of FFNN
3: Fitness (i.e., FFNN error or MSE) can be obtained using Equation 1.
if pBestScore > Fitness **then**
| pBestScore = Fitness and pBestPosition = x.
Else
End
If gBestScore > fitness **then**
| gBestScore = Fitness and gBestPosition = x.
Else
End
4: Now calculate w by using Equation 11.
5: Update velocity $v_{ij}(t+1)$ and position $x_{ij}(t+1)$ of particles according to Equations 11 and 12, respectively.
End
Final: PSO's best particle positions (pBest) are the (***W*** and *b*) for FFNN.

Algorithm 1: Pseudocode for FFNN training using PSO.

particles in the most favorable direction. t denotes the latest version.

W * $v_{i,j}(t)$: explores PSO's new territory, $c1 * r1 * (p_{ij}^{lB}(t) - x_{i,j}(t))$ represents one's own thoughts, and $c2 * r2 * p_j^{gB}(t) - x_{i,j}(t))$ represents particle cooperation.

Algorithm 1 describes our suggested PSO-based parameter selection technique for feedforward NNs, which we have developed and the same returns a configuration that is near optimal. The algorithm is described in terms of two use case scenarios, which are referred to as smart city services and proxy IoT services, respectively. Section 5 will go into further detail on these topics.

5. Results and Discussion

CH selection in IoT may be simulated with MATLAB R2015a. The Xively IoT platform supplies the data that are used in this simulation. In the course of this research, a number of performance indicators, including energy, load, temperature, the number of active nodes, and cost functions, were taken into consideration in order to determine the best CH.

The simulation is based on a field that is 100×100 m in size. The IoT big topic is expected to be at the forefront of the study sector. In the following part of this section, a comparison is made between the results obtained by the suggested model and those obtained by previously developed algorithms such as SVM, GSA, and BPSO algorithms.

5.1. Performance Evaluation: The Number of Alive Nodes. Figure 2 shows how the number of active network nodes is used to evaluate and compare the proposed model to existing models. All nodes are valid during the first 1,000 iterations of the simulation, as shown in the following figure. For all models, the number of active nodes steadily decreased after 1,000 iterations. The current model has no live nodes after 1,700 iterations. Even after 1,700 iterations, the model that was proposed still contained approximately 20 nodes that were active. The suggested model keeps more nodes alive before the final iteration, extending network longevity.

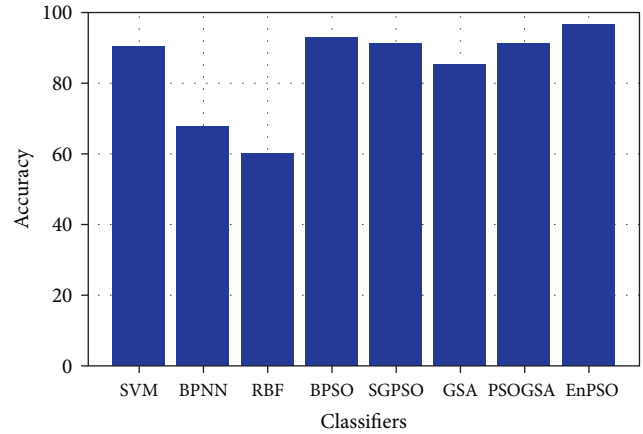


Figure 2: Evaluation of performance established on the total number of functioning nodes.

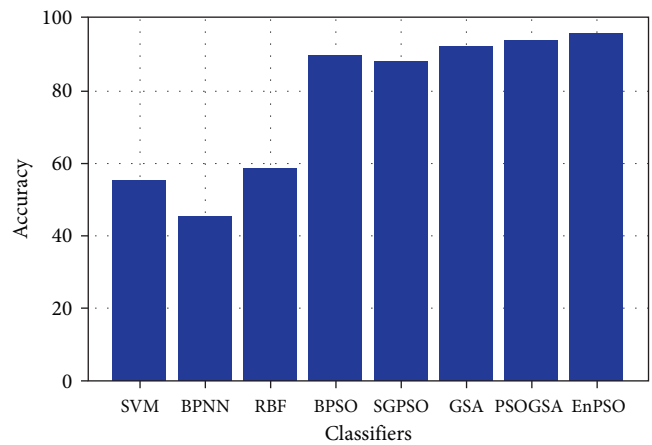


Figure 3: Performance evaluation established on load metric. The temperature produced by the sensor nodes will be lower when the load on the CH is lower.

5.2. Performance Evaluation: The Load Metric. The results of the suggested model with CHs functioning as a load may be shown in Figure 3. If the load is distributed uniformly over all of the CHs, then the efficiency of the IoT network will be at its highest possible level. According to this figure, the load is distributed very evenly over all CHs using the methodology that was provided. In addition to this, the proposed model optimizes the load in each iteration of the process. Even after 1,500 iterations, the CH load is still significantly lower than the version that is currently in use.

The IoT network consumes less energy as a result of this, which increases the network's efficiency.

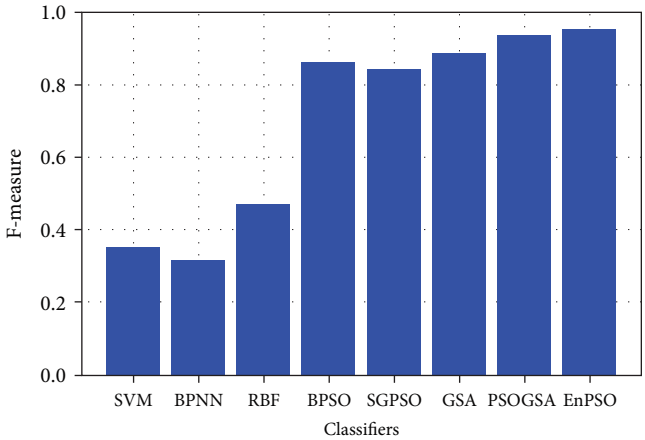


Figure 4: Performance evaluation established on energy metric.

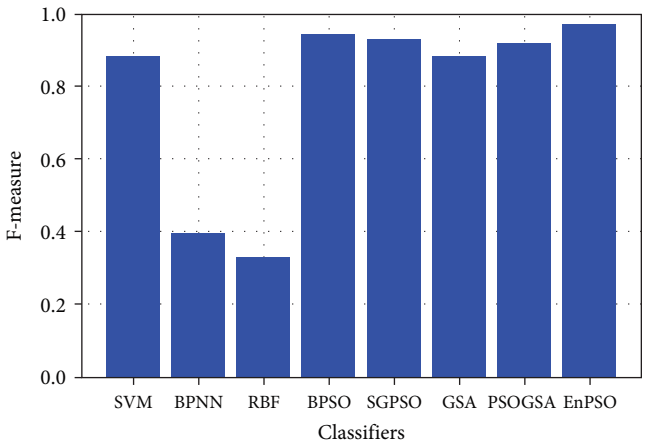


Figure 5: Performance evaluation based on cost function.

5.3. Performance Evaluation: The Energy Metric. The performance of the suggested energy-based model is shown in Figure 4 to the performance of other models that already exist. At the beginning, each model assumes that the energy of the network is 0.5 J. The energy required to complete a task decreases proportionately with the number of iterations performed. In comparison to the model that is currently being used, the proposed model consistently results in a greater amount of energy being present in the network after each iteration. This helps ensure that the network will continue to function effectively in the years to come.

5.4. Performance Evaluation: The Cost Function. Figure 5 depicts the cost function-based output of the proposed model. The convergence of an algorithm is determined by the cost function.

As the number of iterations in an algorithm rises, the convergence of the method should generally get better.

The proposed model outperforms the current model in terms of convergence, as can be seen in the graph. As shown in Figures 2–5, the proposed model outperforms the current model for all the measures considered. Existing models make use of "blind operators" for the sake of manipulation, which are independent of the fitness function. Simulated annealing, which plays the role of an operator in the PSO-NN process, makes it feasible to guarantee that the blind operator is replaced with a local search that uses the solution as the initial state. This makes it possible to guarantee that the blind operator is replaced. After completing the primary objective, the enhanced solution is implemented in place of the initial one. The utilization of the PSO algorithm is consequently improved as a consequence of the process of simulated annealing. As a consequence of this, the simulated annealing algorithm contributes to an improvement in the effectiveness of PSO in locating the best solution. The end conclusion is that the suggested model beats existing approaches when it comes to optimizing the effectiveness of IoT networks.

5.5. Comparison with the State-of-the Art Approaches. Here are some comparisons of hybrid PSO with other approaches in terms of IoT-based smart cities:

(1) Genetic algorithms: Both PSO and genetic algorithms are metaheuristic optimization algorithms that can be used to optimize IoT-based smart cities. However, hybrid PSO combines PSO with genetic algorithms to improve the performance of the algorithm. Hybrid PSO can be more efficient than genetic algorithms alone because it uses PSO to explore the search space and genetic algorithms to exploit the best solutions.

(2) Ant colony optimization: It is another optimization technique that can be used to optimize IoT-based smart cities. Like genetic algorithms, hybrid PSO combines PSO with ant colony optimization to improve the performance of the algorithm. Hybrid PSO can be more efficient than ant colony optimization alone because it uses PSO to explore the search space and ant colony optimization to exploit the best solutions.

(3) Convex optimization: It is a mathematical optimization technique that can be used to optimize IoT-based smart cities. However, convex optimization requires that the objective function and the constraints be convex, which may not always be the case in real-world scenarios. Hybrid PSO does not have this limitation and can be used to optimize a wide range of objective functions and constraints.

(4) Reinforcement learning: It is a machine learning technique that can be used to optimize IoT-based smart cities. However, reinforcement learning requires a large amount of data to train the model, which may not always be available in real-world scenarios. Hybrid PSO does not require training data and can be used to optimize IoT-based smart cities in real time.

In conclusion, hybrid PSO can be a powerful optimization technique for IoT-based smart cities. It combines the strengths of PSO with other optimization techniques to improve the performance of the algorithm and can be used to optimize a wide range of tasks in real time. However, the choice of optimization technique ultimately depends on the specific requirements of the problem at hand.

6. Conclusion and Future Direction

In spite of the fact that the IoT has enormous promise in a large number of different applications in the modern world, there are a lot of hurdles to jump over first. To make the IoT more robust, there are a number of difficulties that need to be resolved, including those pertaining to data access, hardware compatibility, and optimization of power consumption. For the purpose of this investigation, we settled on concentrating on the challenge of energy optimization. In order to solve this problem, this research paper employs a hybrid metaheuristic algorithm known as PSO-NN to optimize the sensor power consumption of this IoT-based WSN. For the purpose of modeling this IoT network, this research makes use of the Xively IoT platform. There have been a total of 2,000 iterations of the IoT. In this piece of work, numerous performance parameters, like cost function, residual energy, number of active nodes, temperature, and load, are used to determine the best CH for the operation of IoT network 370. It then compares the proposed method with various existing methods. The results of the experiments demonstrate that the proposed method is superior to the method that is currently being used. In the future, determining the best CH may also involve taking into account a number of other performance parameters, including link lifetime, node density, and latency. When we utilize this strategy in real-time applications, such as those employed in a variety of sensors, we are also able to test the scalability of the work that we have planned.

Future research work will address the power optimization for each individual clustering IoT networks, and related security issues.

Abbreviations

CH: Cluster head
PSO: Particle swarm optimization
ANN: Artificial neural network
QoS: Quality of service
WSN: Wireless sensor network
BT: Base station
PSO-NN: Particle swarm optimization-neural network.

Data Availability

Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] F. Cicirelli, A. Guerrieri, G. Spezzano, and A. Vinci, "An edge-based platform for dynamic smart city applications," *Future Generation Computer Systems*, vol. 76, pp. 106–118, 2017.

[2] B. N. Silva, M. Khan, and K. Han, "Towards sustainable smart cities: a review of trends, architectures, components, and open challenges in smart cities," *Sustainable Cities and Society*, vol. 38, pp. 697–713, 2018.

[3] M. I Ali, "Multi-layer cross domain reasoning over distributed autonomous IoT applications," *Open Journal of Inter390 Net Of Things (OJIOT)*, vol. 3, no. 1, pp. 75–90, 2017.

[4] S. Soursos, I. P. Zarko, P. Zwickl, I. Gojmerac, G. Bianchi, and G. Carrozzo, "Towards the cross-domain interoperability of IoT platforms," in *2016 European Conference on Networks and Communications (EuCNC)*, pp. 398–402, IEEE, Athens, 2016.

[5] Y. Liu, C. Yang, L. Jiang, S. Xie, and Y. Zhang, "Intelligent edge 400 computing for IoT-based energy management in smart cities," *IEEE Network*, vol. 33, no. 2, pp. 111–117, 2019.

[6] D. S. Roy, R. K. Behera, K. H. K. Reddy, and R. Buyya, "A context aware fog enabled scheme for real-time cross-vertical IoT applications," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 2400–2412, 2018.

[7] A. Bomnale and S. Malgaonkar, "Power optimization in wireless sensor networks," in *2018 International Conference on Communication Information and Computing Technology (ICCICT)*, pp. 1–6, IEEE, New York, 2018.

[8] J. A. Khan, H. K. Qureshi, and A. Iqbal, "Energy management in wireless sensor networks: a survey," *Computers & Electrical Engineering*, vol. 41, pp. 159–176, 2015.

[9] N. Wang, B. Varghese, M. Matthaiou, and D. S. Nikolopoulos, "ENORM: a framework for edge node resource management," *IEEE Transactions on Services Computing*, vol. 13, no. 6, pp. 1086–1099, 2020.

[10] T. H. Luan, L. Gao, Z. Li, Y. Xiang, G. Wei, and L. Sun, "Fog computing: focusing on mobile users at the edge," 2015.

[11] I. Bouzguenda, C. Alalouch, and N. Fava, "Towards smart sustainable cities: a review of the role digital citizen participation could play in advancing social sustainability," *Sustainable Cities and Society*, vol. 50, Article ID 101627, 2019.

[12] S. E. Bibri, "The IoT for smart sustainable cities of the future: an analytical framework for sensor-based big data applications for environmental sustainability," *Sustainable Cities and Society*, vol. 38, pp. 230–253, 2018.

[13] V. Dinesh Reddy, G. R. Gangadharan, and G. S. V. R. K. Rao, "Energy-aware virtual machine allocation and selection in cloud data centers," *Soft Computing*, vol. 23, no. 6, pp. 1917–1932, 2019.

[14] G. Cao, "Topology-aware multi-objective virtual machine dynamic consolidation for cloud datacentre," *Sustainable Computing: Informatics and Systems*, vol. 21, pp. 179–188, 2019.

[15] A. Beloglazov and R. Buyya, "Energy efficient allocation of virtual machines in cloud data centers," in *2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing*, pp. 577-578, IEEE, 2010, May.

[16] R. Dhall and H. Agrawal, "An improved energy efficient duty cycling algorithm for IoT based precision agriculture," *Procedia Computer Science*, vol. 141, pp. 135–142, 2018.

[17] I. Amirtharaj, T. Groot, and B. Dezfouli, "Profiling and improving the duty-cycling performance of Linux-based IoT devices," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–29, 2019.

[18] B. Afzal, S. A. Alvi, G. A. Shah, and W. Mahmood, "Energy efficient context aware traffic scheduling for IoT applications," *Ad Hoc Networks*, vol. 62, no. 62, pp. 101–115, 2017.

[19] A. Kozłowski and J. Sosnowski, "Energy efficiency trade-off between duty-cycling and wake-up radio techniques in IoT networks," *Wireless Personal Communications*, vol. 107, no. 4, pp. 1951–1971, 2019.

[20] C. Moser, L. Thiele, D. Brunelli, and L. Benini, "Adaptive power management in energy harvesting systems," in *2007*

Design, Automation Test in Europe Conference Exhibition, pp. 1–6, IEEE, 2007.

[21] C. M. Vigorito, D. Ganesan, and A. G. Barto, "Adaptive control of duty cycling in energy-harvesting wireless sensor networks," in *2007 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks*, pp. 21–30, IEEE, 2007.

[22] J. Hu, Q. Qian, A. Fang, S. Wu, and Y. Xie, "Optimal data transmission strategy for healthcare-based wireless sensor networks: A stochastic differential game approach," *Wireless Personal Communications*, vol. 89, no. 4, pp. 1295–1313, 2016.

[23] J. Hsu, S. Zahedi, A. Kansal, M. Srivastava, and V. Raghunathan, "Adaptive duty cycling for energy harvesting systems," in *Proceedings of the 2006 international symposium on Low power electronics and design*, pp. 180–185, IEEE, 2006.

[24] R. Jurdak, P. Corke, D. Dharman, and G. Salagnac, "Adaptive GPS duty cycling and radio ranging for energy-efficient localization," in *Proceedings of the 8th ACM Conference on Embedded Networked Sensor Systems*, pp. 57–70, 2010.

[25] D. Lee and K. Chung, "Adaptive duty-cycle based congestion control for home automation networks," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 1, pp. 42–47, 2010.

[26] S. R. Barath, B. U. Usha, G. B. Arpitha, V. Kumar, and S. M. Yaseen, "Strategies for Congestion Control in Wireless Sensor Networks: A Survey," in *2019 International Conference on Smart Systems and Inventive Technology (ICSSIT)*, pp. 629–633, IEEE, 2019.

[27] C. Zhan and H. Lai, "Energy minimization in Internet-of-Things system based on rotary-wing UAV," *IEEE Wireless Communications Letters*, vol. 8, no. 5, pp. 1341–1344, 2019.

[28] S. Sendra Compte, J. L loret, M. Garcıa Pineda, and J. F. Toledo Alarcon, "Power saving and energy optimization techniques for wireless sensor networks," *Journal of Communications*, vol. 6, no. 6, pp. 439–459, 2011.

[29] S. S. Beheshtiha, H. P. Tan, and M. Sabaei, "Opportunistic routing with adaptive harvesting-aware duty cycling in energy harvesting WSN," in *The 15thInternational Symposium on Wireless Personal Multimedia Communications*, pp. 90–94, IEEE, 2012.

[30] O. Briante, A. M. Mandalari, A. Molinaro, G. Ruggeri, J. AlonsoZarate, and F. Vazquez-Gallego, "Duty-cycle optimization for machine-to-machine area networks based on frame slotted-ALOHA with energy harvesting capabilities," in *European Wireless 2014; 20th European Wireless Conference*, pp. 1–6, IEEE, 2014.

[31] C. H. S. Oliveira and M. D. Castro, "Autonomic Duty Cycling for Target Tracking in a Bio-Inspired Wireless Sensor Network," in *International Conference on Systems (ICONS'15)*, pp. 112–118, ICONS.

[32] C. H. Oliveira, Y. Ghamri-Doudane, and S. Lohier, "A duty cycle self-adaptation algorithm for the 802.15. 4 wireless sensor networks," in *Global Information Infrastructure Symposium-GIIS. 2013*, pp. 1–7, IEEE, 2013, October.

[33] J. Tang, H. Tang, X. Zhang et al., "Energy minimization in D2D-Assisted Cache500 enabled internet of things: a deep reinforcement learning approach," *IEEE Transactions on Industrial Informatics*, 2019.

[34] S. Vergis, V. Komianos, G. Tsoumanis, A. Tsipis, and K. Oikonomou, "A low-cost vehicular traffic monitoring system using fog computing," *Smart Cities*, vol. 3, no. 1, pp. 138–156, 2020.

[35] K. Ma, A. Bagula, C. Nyirenda, and O. Ajayi, "An IoT-based fog com510 putting model," *Sensors*, vol. 19, no. 12, Article ID 2783, 2019.

[36] O. Skarlat, M. Nardelli, S. Schulte, and S. Dustdar, "Towards qos-aware fog service placement," in *2017 IEEE 1st international conference on Fog and Edge Computing (ICFEC)*, pp. 89–96, IEEE, 2017.

[37] V. B. Souza, X. Masip-Bruin, E. Marín-Tordera et al., "Towards a proper service placement in combined Fog-to-Cloud (F2C) architectures," *Future Generation Computer Systems*, vol. 87, pp. 1–15, 2018.

[38] Q. T. Minh, D. T. Nguyen, A. Van Le, H. D. Nguyen, and A. Truong, "Toward service placement on Fog computing landscape," in *2017 4th NAFOSTED conference on information and computer science*, pp. 291–296, IEEE, 2017.

[39] M. I. Naas, P. R. Parvedy, J. Boukhobza, and L. Lemarchand, "iFogStoran IoT data placement strategy for fog infrastructure," in *2017 IEEE 1st International Conference on Fog and Edge Computing (ICFEC)*, pp. 97–104, IEEE, 2017.

[40] M. Taneja and A. Davy, "Resource aware placement of IoT application modules in Fog-Cloud Computing Paradigm," in *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, pp. 1222–1228, IEEE, 2017.

[41] I. Lera, C. Guerrero, and C. Juiz, "Availability-aware service placement policy in fog computing based on graph partitions," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3641–3651, 2019.

[42] K. A. Eldrandaly, M. Abdel-Basset, and L. Abdel-Fatah, "PTZ-Surveillance coverage based on artificial intelligence for smart cities," *International Journal of Information Management*, vol. 49, pp. 520–532, 2019.

[43] R. K. Behera, K. H. K. Reddy, and D. S. Roy, "A novel context migration model for fog-enabled cross-vertical IoT applications," in *International Conference on Innovative Computing and Communications*, pp. 287–295, Springer, Singapore, 2020.

[44] K. H. K. Reddy, R. K. Behera, A. Chakrabarty, and D. S. Roy, "A service delay minimization scheme for QoS-constrained, context-aware unified IoT applications," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 10527–10534, 2020.

[45] O. Skarlat, M. Nardelli, S. Schulte, M. Borkowski, and P. Leitner, "Optimized IoT service placement in the fog," *Service Oriented Computing and Applications*, vol. 11, no. 4, pp. 427–443, 2017.

[46] M. Zubair, N. Javaid, M. Ismail, M. Zakria, M. A. Zaheer, and F. Saeed, "Integration of cloud-fog based platform for load balancing using hybrid genetic algorithm using bin packing technique," in *International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, pp. 279–292, Springer, 2018.

[47] Y. Sun, F. Lin, and H. Xu, "Multi-objective optimization of resource scheduling in fog computing using an improved NSGA-II," *Wireless Personal Communications*, vol. 102, no. 2, pp. 1369–1385, 2018.

[48] C. Guerrero, I. Lera, and C. Juiz, "Evaluation and efficiency comparison of evolutionary algorithms for service placement optimization in fog architectures," *Future Generation Computer Systems*, vol. 97, pp. 131–144, 2019.

[49] A. Kaur and S. K. Sood, "Cloud-Fog based framework for drought prediction and forecasting using artificial neural network and genetic algorithm," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 32, no. 2, pp. 273–289, 2020.

[50] A. Gyrard, S. K. Datta, C. Bonnet, and K. Boudaoud, "Cross domain Internet of Things application development: M3 framework and evaluation," in *2015 3rd International Conference on Future Internet of Things and Cloud*, pp. 9–16, IEEE, 2015.

Hindawi
Mathematical Problems in Engineering
Volume 2023, Article ID 1110500, 11 pages
https://doi.org/10.1155/2023/1110500

Research Article

Prediction of Alzheimer's Disease Using DHO-Based Pretrained CNN Model

S. Venkatasubramanian,[1] **Jaiprakash Narain Dwivedi**,[2] **S. Raja**,[3] **N. Rajeswari**,[4] **J. Logeshwaran**,[5] **and Avvaru Praveen Kumar**[6]

[1]*Department of Computer Science and Business Systems, Saranathan College of Engineering, Trichy 620012, Tamilnadu, India*
[2]*Electronics & Communication Engineering, School of Engineering & Technology, Lingaya's Vidyapeeth, Faridabad, Haryana, India*
[3]*Managing Director, Research & Development, Mr. R BUSINESS CORPORATION, Karur, Tamilnadu, India*
[4]*Department of Mechanical Engineering, Surya Engineering College, Erode, Tamilnadu, India*
[5]*Department of Electronics and Communication Engineering, Sri Eshwar College of Engineering, Coimbatore 641202, Tamilnadu, India*
[6]*Department of Applied Chemistry, School of Applied Natural Science, Adama Science and Technology University, P.O. Box 1888, Adama, Ethiopia*

Correspondence should be addressed to Avvaru Praveen Kumar; avvaru.praveen@astu.edu.et

Received 24 January 2023; Revised 8 May 2023; Accepted 17 May 2023; Published 6 June 2023

Academic Editor: Ardashir Mohammadzadeh

Detecting Alzheimer's disease (AD) early on allows patients to take preventative measures before the onset of irreversible brain damage, which is a critical factor in the treatment of Alzheimer's patients. Most machine detection methods are constrained by congenital observations, although computers have been utilized in several recent research studies to diagnose AD. In AD, the hippocampus is usually the first part of the brain to be affected. Structural magnetic resonance imaging (SMRI) can be used to assist in diagnosing AD by measuring the hippocampus's form and volume (MRI). The information encoded by these attributes is restricted and may be affected by segmentation problems. These traits are also extracted independently of the classification, which could result in lower-than-desired classification accuracy. Researchers in this study used structural MRI data to develop a deep learning framework for combined automatic hippocampus segmentation and AD categorization. Multi-task deep learning (MTDL) is used to learn hippocampus segmentation simultaneously. The hyperparameter optimization of the CNN model (capsule network) for illness classification is then carried out using the deer hunting optimization (DHO) technique. ADNI-standardized MRI datasets have been used to test the suggested method, and it is accurate. Suggested MTDL achieved 97.1% accuracy and 93.5% of Dice coefficient, whereas the proposed MTDL model achieved an accuracy of 96% for binary classification and 93% for multi-class classification.

1. Introduction

Alzheimer's disease is a brain ailment that gradually impairs thinking and memory abilities as well as the capacity to do even the most basic tasks. An intracellular protein called cAMP-response element binding protein (CREB) controls the expression of key genes in dopaminergic neuron [1]. The shared form of dementia, AD, poses a significant test to healthcare providers in the twenty-first century. In the United States, 5.5 million people who are 65 years and older have AD, making it the sixth greatest mortality [2]. In 2018, the total cost of controlling AD in the United States was $277 billion, with a significant impact on the broader economy and a strain on the country's healthcare system [3]. In the absence of a treatment that has been proven to alter the course of the disease, a considerable deal of work has been put into developing procedures for early identification, particularly in presymptomatic phases [3]. Advances in

neuroimaging techniques, including MRI and PET, have been utilized to discover AD-related structural and molecular biomarkers [4]. Brain imaging technology has progressed at an incredible rate, making it difficult to incorporate enormous amounts of high-dimensional multimodal data. Computer-aided machine learning methodologies for integrative analysis have become increasingly popular as a result. AD progression can be predicted using well-known pattern analysis approaches such as LPBM, logistic regression, and support vector machine (SVM) [5].

Preprocessing or architectural design is required to use these machine learning techniques [6]. Dimensionality reduction is a common aspect in machine learning classification investigations, as is the extraction and selection of features, as well as the selection of classification methods based on features. These techniques necessitate a high level of specialized knowledge and may take a long time to optimize through numerous phases. A problem has arisen in the reproducibility of these methods [7, 8]. Neuroimaging modalities can be used to pick AD-related features in the feature selection process, brain glucose metabolism, and amyloid buildup in research regions (ROIs), such as the hippocampus, such as mean subcortical volumes, densities of grey matter, and cortical thickness [9, 10].

It is becoming more and more common for large-scale medical imaging analysis to use "on-the-fly" deep (or "on-the-fly") learning to generate features from raw neuroimaging data [11]. Deep learning techniques for AD diagnosis are based on short MRI datasets, which makes it difficult for researchers to build deep CNN models with a significant number of parameters that must be learned [12, 13].

1.1. Problem Statement. Hippocampal analysis methods now in use have several flaws. First, precise segmentation of the hippocampus is required for both hippocampal volumetric and shape analyses. The hippocampus is difficult to correctly segment because of its irregular shape and unclear boundary in MRI. Handcrafted shapes may not be suitable for examination in the future, affecting categorization performance in the diagnosis of illnesses. According to a third study, the hippocampus alone may not be sufficient to distinguish mild cognitive impairment (MCI) patients from healthy controls. In the early stages of AD, both the amygdala and the para-hippocampus are also affected by the condition. As the last point, MRI images taken from the hippocampal region can be very helpful in the diagnosis of AD.

1.2. Contribution. Machine learning/deep learning algorithms have been used to detect biomarkers and interpret illness aetiology in recent years. Detecting AD can be done in a variety of ways, including analyzing MRI images for specific areas of interest (ROIs). The hippocampus is an essential anatomical region in the pathogenesis of AD since it is one of the first brain ROIs to be impacted. A new deep learning framework combining an MTDL model and an MTDL model for simultaneous hippocampus segmentation and illness organization using MRI data is suggested to address the aforementioned issues listed in the problem description.

2. Related Works

Faisal and Kwon's goal [14] was to design a deep learning system that could extract useful AD biomarkers from physical MRI and classify brain pictures into AD, MCI, and CN groups. In this study, researchers used ADNI datasets available online to train CNNs on MRI brain pictures. It was used to merge features from multiple into compact high-level features by using our proposed process. Using the proposed method, computation time is lowered because there are fewer variables to deal with. Comparative evaluations of our suggested convolution operation vs. the most extensively used AD classification metrics, such as accuracy and area under ROC curve (AUC), are performed.

Early detection of various phases of cognitive impairment and AD utilizing neuroimaging and transfer learning (TL) was the emphasis of Shanmugam et al. [15]. Images from ADNI's database with varied CN, early mild cognitive impairment, moderate cognitive impairment, and late-MCI as LMCI classifications are classified using transfer learning. There are three pretrained networks utilized in this categorization that have been trained and evaluated on 6000 photos from the ADNI collection. Confusion matrices and their properties are used to evaluate the classification presentation of the three networks. GoogLeNet, AlexNet, and ResNet-18 all have an overall accuracy of 96.39%, 94.08%, and 97.51%, respectively, in detecting Alzheimer's disease. Confusion matrix parameters were also used to examine the pretrained networks' performance within classes.

There are numerous techniques to utilize deep learning classification to categorize Alzheimer's disease, according to Samhan et al. [16]. In large trials, adopting this method will result in better patient care and lower costs. Python was used in the development of the system, which is particularly useful for doctors in the classification of AD. 70% of the image was used to train the model, and 30% was used to verify it. On a series of held-out tests, our trained model was 100% accurate.

As a potential tool for identifying people with AD-related dementia, Tian et al. [17] investigated the retina, specifically the retinal vascular system. Adding a saliency analysis on top of the high level of classification accuracy helps make this pipeline easier to understand. Saliency study shows that retinal images with small vessels provide more information for Alzheimer's disease diagnosis than images with large vessels.

To classify this chronic condition as AD, Divya and ShanthaSelvaKumari [18] employed several feature selection strategies and distinct classifiers. When the number of records with large dimensions is few, it is much easier to classify those records. They yielded accuracy rates of 968.22%, 89.59%, and 90.40% after several attempts to pick the best features. SVM with radial basis function kernel yielded these higher accuracy rates. In the MCI/AD classification, a 2.7% improvement in the MMSE score was seen,

but it had no impact on the NC/AD and NC/MCI classifications.

"The wisdom of experts" can be harnessed by using An et al.'s deep ensemble learning framework [19] to integrate multi-source data. Training two sparse autoencoders for feature learning at the voting layer helps to minimize the connection between characteristics and diversify the base classifiers. Classifiers are ranked using a deep belief network that uses a nonlinear feature-weighted algorithm at the stacking layer, which may violate conditional independence. As a sort of meta-classifier, the neural network is employed. To deal with a cost-sensitive issue, oversampling and threshold shifting are employed at the optimization layer. An ensemble of probabilistic predictions is combined with a similarity computation to produce optimized forecasts. Alzheimer's illness is classified using the new deep ensemble learning framework. Our proposed framework outperforms six well-known ensemble techniques, including the classic stacking algorithm, in classification accuracy tests using clinical data.

Densely linked convolutional neural networks with connection-wise attention mechanisms were proposed by Zhang et al. [20] to learn the properties of brain MR images for AD classification. Pictures are preprocessed using a dense CNN, which extracts multi-scale features, and a connection-wise attention mechanism is utilized to integrate connections among features from diverse layers to turn the MR images into more compact high-level features. MRI's spatial information can be captured by extending the convolution operation to 3D. All of the previous layers' features were combined with those from the 3D convolution layer in various ways before being used to classify the data. Based on baseline MRI scans of 968 ADNI database participants, The authors tested the technique to distinguish between AD and healthy patients, MCI converters and healthy subjects, and MCI using MCI scans.

2.1. Challenges in Brain MRI Segmentation

(i) Brain structural structures differ greatly among individuals due to genetics, age, gender, and illness. Using a single segmentation algorithm across all phenotypic subgroups is problematic.

(ii) For example, it is difficult to deal with cytoarchitectural changes such as the thickness of tissue, the depth of the sulci, and smooth boundaries between tissue types. This might lead to a muddled categorization of various tissue types. Even human professionals have difficulty with this.

(iii) These modalities have a low contrast of anatomical structure, which leads to poor segmentation performance.

(iv) Manual segmentation is tedious and subjective and requires a deep understanding of brain anatomy to perform. Thus, it is challenging to acquire sufficient data for creating a segmentation model.

(v) In an ordinary image for segmentation, the noisy backdrop makes it difficult to apply an appropriate label to each pixel/voxel with learned characteristics.

(vi) In addition to its tiny size and volume, the hippocampus is one of the most important biomarkers for AD because of its structural heterogeneity, partial volume effects and low contrast, and low signal-to-noise ratio.

3. Proposed Model

One of the ways to diagnose AD is represented in Figure 1. The MRI slices must be obtained initially. Preprocessing removes irrelevant information from the data and reorients them so that they can be interpreted more easily. The preprocessed data are segmented using deep learning to retrieve the properties from the brain MRI. For example, a classifier uses parameters like the patient's body surface area, the center of gravity, intensity, and standard deviation to determine whether he or she is developing AD or not.

3.1. Dataset Description. MCI and early-onset Alzheimer's disease can be tracked using MRI, PET scans, and other biomarkers as part of the Adverse Childhood Neuropsychiatric Disorders Initiative (ADNI). Written informed consent for the collection of imaging and genetic samples was signed by the subjects at the time of enrolment and approved by the Institutional Review Boards (IRBs) at each participating location.

A total of 449 participants were randomly selected to participate in the study. MMSE stands for Mini-Mental State Examination, and CDR stands for Clinical Dementia Rating. For 1.5 T MR imaging, we used images obtained by the ADNI acquisition method [21]. Image acquisition procedures are explained in greater detail on the ADNI website. Images are resized to the dimension of 11 cubic mm to fit on a single sheet of paper. As a result of this treatment, their skulls were scraped and their cerebellums were removed. The FMRIB Software Library (FSL) 5.0 from https://fsl.fmrib.ox.ac.uk/ was utilized in this project and used a template image with 12 degrees of freedom and a set of evasion parameters to align all MR pictures.

ADNI participants' demographic and clinical information is shown in Table 1 (mean standard deviation). AD, mild cognitive impairment, and normal control are all referred to as "AD," "MCI," and "NC," respectively.

3.2. Preprocessing. Nonlinear gradients in a picture can distort an image using a method called Gradwarp [22]. Gradient models have a different kind of nonlinearity. The geometrical features of an image can be tweaked to improve its information. B1 nonuniformity is used to rectify image color and intensity information because of mishandled radio frequency transmission. N3 bias field correction corrects the distortion caused by dielectric effects during acquisition [23]. Although N3 bias field correction is used for 1.5 T images to improve the nonuniform gradient in the image, these effects are widespread in 3 T machines. Before the N3 correction, Gradwarp and B1 corrections have been applied.

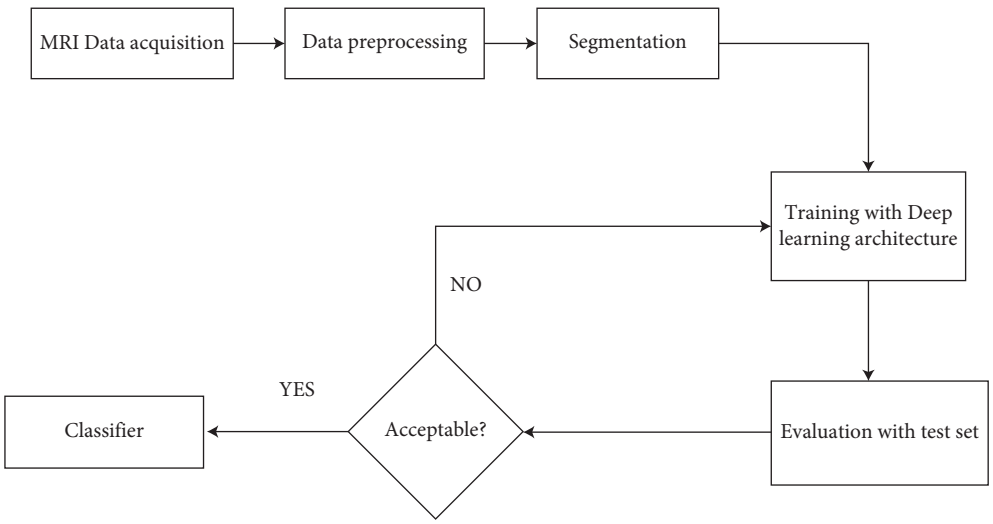


Figure 1: The overall block diagram of AD diagnosis.

Table 1: Data description.

Diagnosis	Age	MMSE	Education (year)	Gender (M/F)	CDR
AD	75.9 ± 6.8	23.2 ± 1.8	15.0 ± 3.0	48/49	0.8 ± 0.3
NC	75.9 ± 5.0	29.2 ± 1.0	15.6 ± 2.7	59/60	0 ± 0
MCI	75.2 ± 7.3	26.9 ± 1.8	15.7 ± 2.6	145/88	0.5 ± 0

MMSE, Mini-Mental State Examination; CDR, Clinical Dementia Rating.

Image segmentation has been used in the literature to improve classification accuracy [24, 25]. These images were preprocessed with the use of the segmentation module of statistical parametric mapping (SPM) available at http://www.fiiion/ion-ucl/spm. To map MRI scans onto tissue probability maps, SPM uses these maps to extract the mapped regions. The MRI scan is segmented into three parts using bias correction and normalization in this module. The output of mapping can be linked to the orientation of a picture using a process called registration. Brain registration is all about minimizing the impact of external elements like the scalp on the segmented pictures of the cerebral cortex that are generated.

3.3. Segmentation Using MTDL for Joint hippocampus. In the human brain, there is a small area known as the hippocampus in the medial temporal lobe. The hippocampus contains a disproportionately small number of voxels compared to the rest of the brain, resulting in a very unbalanced dataset. After preprocessing and registration, the next step is to create 3D image patches with hippocampus-specific bounding cubes. The 3D axes of the bounding cubes are used to extract 3D patches from MR images. It is important to consider the size of the bounding cube when determining how the hippocampus is segmented. A large bounding cube may also lead to the class imbalance problem, increasing the computation time. Small bounding cubes can impede the segmentation of the hippocampus. An empirical study found that a voxel bounding cube of $64 \times 48 \times 64$ voxels was the optimal size for the trade-off. The patches form the basis of our deep learning model for segmenting the hippocampus and classifying illnesses.

Jointly learning hippocampus segmentation and an illness classification is a novel approach that differs from standard methods in which these two tasks are performed separately. To classify images and identify objects, researchers frequently utilize CNNs. V-Net, a volumetric and complete CNN for prostate segmentation in MRIs, has been proposed. This is a multi-task deep CNN model for joint hippocampus segmentation and illness classification inspired by the success of V-Net in prostate segmentation.

Residual functions are learned at convolutional stages using a deep CNN, which aims to achieve fast convergence. "ResNet Block 1" and "ResNet Block 2" are two residual blocks, each consisting of 3D convolution, batch normalization (BN), parametric rectified linear unit (PReLU) activation, and dropout layers, as illustrated in Figure 2. The input is added to the output of the second convolutional layer to learn a residual function in ResNet Block 1. For each block, the input is added to the outputs of both convolutional layers for a residual function, which is learned in Block 2 of ResNet. There are batches of MRI data that are used to train the kernels. Fast inference is easier to achieve with small kernels since there are fewer parameters to train. More complicated patterns and greater expressiveness can be learned by larger kernels. Layers of tiny kernels can be stacked to generate this appearance. For all convolutions, the kernel size is fixed at $3 \times 3 \times 3$. A nonlinear PReLU activation is used to activate the learned filters, and a feature map is then constructed for each one.

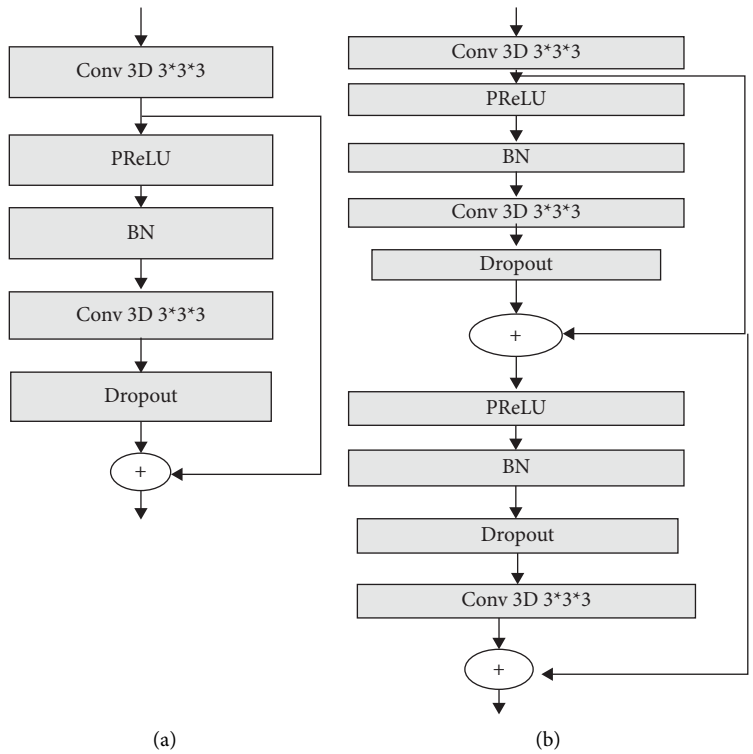


Figure 2: The network architecture of (a) ResNet Block 1 and (b) ResNet Block 2, consisting of 3D convolution, PReLU, BN, and dropout layers.

Downsampling is used to minimize the size of feature maps and improve the receptive field of features in the following layers during the compression stage. Using convolution with kernels of size $2 \times 2 \times 2$ and stride 2, it is implemented. A volumetric segmentation mask is generated by expanding the spatial support of the lower resolution feature maps during the decompression step. The 222 and stride 2 kernels are used for the upsampling via deconvolution. For the probabilistic segmentation of the hippocampus regions, the outputs are transformed to voxel-wise softmax by applying a convolutional layer with a 111 kernel and stride 1. As the last step, the probability output is converted into a binary mask by setting the threshold to 0.

Optimizing the Dice loss function, which measures how well our model can separate hippocampus voxels from the background, is the goal for subject m's hippocampal segmentation:

$$L_s^m = 1 - \frac{2\sum_{i=1}^{N} p_i q_i + \in}{\sum_{i=1}^{N} p_i^2 + \sum_{i=1}^{N} q_i^2 + \in}. \quad (1)$$

If the numerator is zero, a little number is denoted to avoid the numerator from being zero. This is done by using the segmentation prediction (p_i) and the ground truth label (qi). If the number of foregrounds and background voxels is sufficiently unbalanced, the Dice loss function can be utilized. Fully linked layers are used as decompression components to increase classification accuracy. Comparing the predicted and actual labels for subject m, we utilize the categorical cross-entropy loss.

$$L_C^m = -\left(\widehat{y_m} \log y_m + \left(1 - \widehat{y_m}\right) \log\left(1 - y_m\right)\right). \quad (2)$$

The multitasking deep loss function segmentation loss and classification loss are added together to create the CNN model:

$$L_M = a.L_s + (1 - a).L_C = \frac{1}{M} \prod_{m=1}^{M} \cdot \left[a.L_s^m - (1 - a).\left(\widehat{y}_m \log y_m + \left(1 - \widehat{y_m}\right) \log \ \left(1 - y_m\right)\right)\right]. \quad (3)$$

The total number of subjects is M; y_m and y_m are the ground truth label and the anticipated label for subject m, respectively. Losses in hippocampus segmentation and illness classification training are taken into account by weighting the parameter a = [0, 1]. Classification is more critical than segmentation in the early stages of training for a multi-task deep CNN model. Initial warm-up emphasizes segmentation by setting a value of 1 for a. After that, it goes down to 0.5 for training in multi-tasking. Finally, a is set to 0 so that the classification process can take precedence. The Adam approach is utilized to jointly optimize the multi-task network model, and a backpropagation algorithm is used to calculate the network gradients.

After correcting the hippocampal segmentation findings, the hippocampal image patches are shown. Before and after manual corrections, the mean, standard deviation, and range of hippocampus volumes are shown for several groups of participants. After adjustment, we can see that the mean and

SD of hippocampus volume have decreased. For AD, MCI, and NC patients, Figure 3 depicts the scatterplots [21, 22].

3.4. Classification. Sabour et al. [25] were able to overcome the limitations of CNN by employing a higher-dimensional vector known as a "capsule" to represent an entity rather than an individual neuron. The properties of a specific entity portrayed in an image are reflected in the neuronal activity of the active capsule. These features, including the likelihood and a set of parameters such as albedo (color), hue (texture), or deformation (deformation), are taught to a capsule for each visual item. An entity's attributes and the likelihood of existence are represented in CapsNet's input and output as vectors with direction and norm. The model is used to improve forecasts of AD by predicting a high-level capsule's instantiation parameter over a conversion matrix by employing similar levels of capsules. The natural logarithm base, e, is used to define the spiral shape as a constant. To evaluate it, one may use

$$\overrightarrow{P}_s(x) = \left(\overrightarrow{D}_s \times x' \times y' \times z'\right) + \overrightarrow{P}_{bs}(x). \tag{4}$$

It is used to save the best solutions and boost the position of a separate search agent, for example, using $\overrightarrow{P}_s(x)$. The DHO presented here begins with a random sample of the population. The search agent might move closer or farther away from the ideal search agent as it iterates. To ensure that the shift from exploitation to exploration goes well and it is in control, the DHO becomes a global optimizer when it has a strong exploitation and exploration capacity.

3.4.1. DHO-Based Hyperparameter Tuning. A new metaheuristic DHO approach based on deer hunting was developed by a group of hunters for the tuning of hyperparameters. Hunters employ a variety of strategies to surround and approach the deer as closely as possible when hunting it. Deer position and wind angle have to be taken into account when using this technique. Another crucial element of successful hunting is a sense of camaraderie among the participants. Following their successor and leader, their final goal is achieved. The graphic below depicts the model's goal function:

$$f(x) = \max\ (\text{accuracy}). \tag{5}$$

When it comes to weight loss, the DHO method relies on the deer's unique abilities to elude hunters. A haphazard gathering of hunting vectors catalyzes the process. It is described using the following equation:

$$X = \{X_1, X_2, \ldots, X_m\} 1 < j \leq m. \tag{6}$$

There are two ways to express how much population (or "weight") a hunter has when optimizing his strategy. Next, important elements like weight, position, and wind angle are used. Because the entire search area is considered a circle, it is possible to define the wind angle as the circle's diameter.

$$\theta_j = 2\pi a, \tag{7}$$

where a stands for the arbitrary value within the range = [0, 1] and J stands for the current iteration. The location propagation for optimization with the leader position (X_l) and succeeding position (X_s) is provided. The placement of the following weights is determined by the successor location, whereas the primary location of the hunter is determined by the leader location.

X_l is used to spread the message. Everyone tries to reach the optimal location after establishing an optimal location. To begin updating the location, we simulate the surrounding behavior as shown below:

$$X_{j+1} = X_l - Y.p.\left|L \times X_l - X_j\right|. \tag{8}$$

The current iteration's location is designated as X_j, whereas the location for the next iteration is designated as X_{j+1}. This process is aided by the Z and K coefficient vectors. If wind speed is taken into account, an arbitrary value of p can be generated, and this number ranges from 0 to 2. The Z and K coefficient vectors can be estimated using the expressions below:

$$\begin{aligned} Z &= \frac{1}{4}\log\left(j + \frac{1}{j_{\max}}\right)b, \\ K &= 2.c, \end{aligned} \tag{9}$$

where $j_{\max}$ is the maximum iteration. In addition to the range [0, 1], the value of the b variable ranges from −1 to 1. (X, Y) is the initial location of the hunter, which gets upgraded based on the location of the prey. $X\ b$ and $Y\ b$ are recalculated using the Z and K coefficient vectors. When the value of p is less than 1, a position update procedure takes place that allows the hunter to move in any direction without regard to the angle. Transmission utilizies a slanted inclination. Search space is expected to expand as a result of the angle location updation. The angle of the hunter's position is critical to the success of the hunting strategy. To put this into action, consider

$$X_{j+1} = X_l - p.\left|\cos(v) \times X_l - X_j\right|. \tag{10}$$

The ideal position can be shown as $B = (j + 1), X\ (b\ j)$, and p, where p signifies the arbitrary values. The angle location is opposite to the individual location, so the prey does not have any sense of the hunter's presence via the successor location. The vector K is shown within the encircling behavior in the exploration. K values are first considered to be less than 1 to perform an arbitrary search. As a final point, a successor location is used instead of the best possible location in the location updating method. As a final step, a worldwide search is conducted.

$$X_{j+1} = X_s - Z.p\left|K \times X_s - X_j\right|. \tag{11}$$

Site updates are carried out so that an ideal location can be found (namely, termination condition). By optimizing

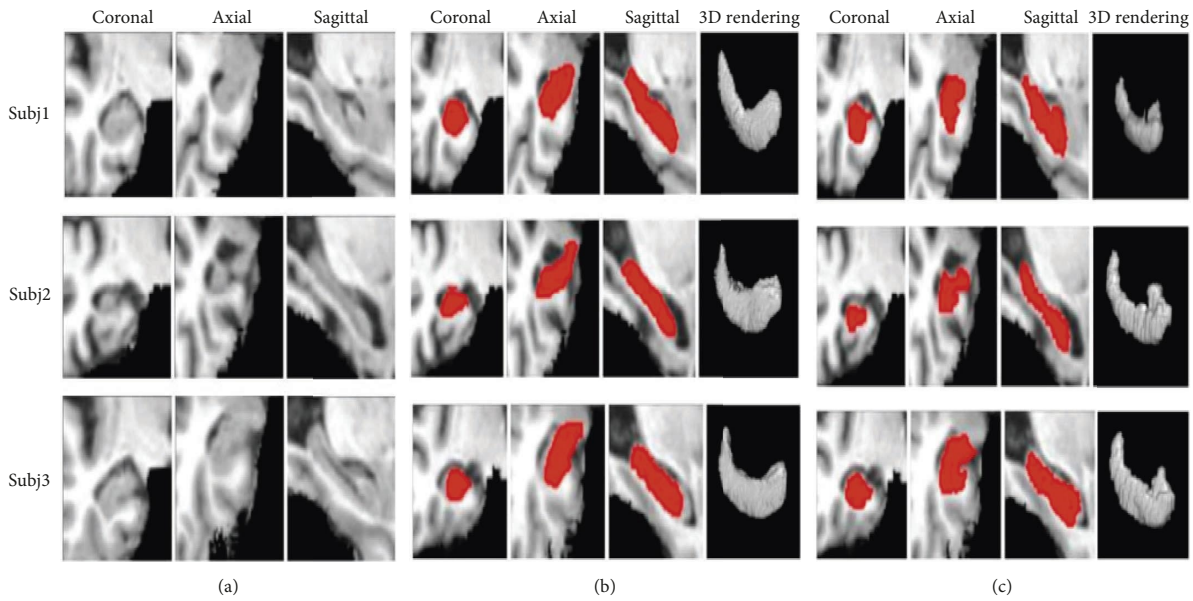


Figure 3: Hippocampal image patches: (a) without patches segmentation labels, (b) patches overlaid with segmentation labels before manual correction, and (c) segmentation tags added to the patches following manual adjustment. Subj1, Subj2, and Subj3 are three people from each of the three different study groups which were chosen at random [21, 22].

the weight parameters of the pretrained CNN model, it is effectively used to identify whether the patient is AD or normal, and the multi-class classification of AD is also performed.

4. Results and Discussion

To create the segmentation and classification model, a high-level neural networks API with Tensorflow as the backend was employed. Keras was used because of its ease of use and ability to run on a GPU.

4.1. Evaluation Metrics. Our method's segmentation and classification performance is assessed using the challenge evaluation measures such as accuracy (AC), Jaccard index (JSI), and Dice coefficient (DSC) in segmentation analysis. AC, specificity (SP), and sensitivity (SE) are all part of the classification's evaluation process. The criteria for evaluating performance are laid forth as follows:

$$\begin{aligned}
\mathrm{SE} &= \frac{tp}{tp + fn}, \\
\mathrm{SP} &= \frac{tn}{tn + fp}, \\
\mathrm{AC} &= \frac{tp + tn}{tp + fp + tn + fn}, \\
\mathrm{DSC} &= \frac{2TP}{(2TP + FP + FN)}, \\
\mathrm{JSI} &= \frac{TP}{FP + FN + TP}, \\
tp_{\mathrm{rate}} &= \frac{tp}{p}, \\
tn_{\mathrm{rate}} &= \frac{tn}{p},
\end{aligned} \tag{12}$$

where tp, tn, fp, and fn denote the number of true positive, true negative, false positive, and false negative.

4.2. Comparative Analysis for Proposed Segmentation (MTDL). In this section, the proposed model is compared with existing techniques such as fuzzy c-means (FCM), adaptively regularized kernel FCM (ARKFCM), and fast and robust FCM (FRFCM). Table 2 and Figure 4 provide the experimental analysis for MTDL with existing models [26].

In Table 2, the analysis represents the validation results for different segmentation techniques. In the first method, FCM achieved an accuracy of 84.8% and the next FRFCM achieved an accuracy of 92.6%, and this accuracy performance is better than FCM. ARKFCM reached the accuracy percentage of 96.5%. Finally, the proposed MTDL reached a better accuracy of 97.1% and achieved better performance than other methods. In the analysis of DSC, FCM achieved 82.1%, FRFCM achieved 91%, ARKFCM achieved 92% and the proposed model achieved 93.5%. Finally, JSI is high for MTDL (i.e., 87.8%) compared to existing FCM models (69% to 85%).

4.3. Comparative Analysis of Proposed Classification. Two types of analysis such as binary classification (normal or AD) and multi-class classification (AD/MCI/NC) are carried out, where Table 3 and Figure 5 show the experimental analysis of the proposed classifier with existing techniques. For better performance, all techniques are implemented with DHO.

Table 3 represents the comparative analysis of binary classification of different models such as RNN, recurrent neural network, and CapsNet. The classifier model of the recursive neural network reached a sensitivity of 93.00% and an accuracy of 90.00%. The recurrent neural network model reaches an accuracy of 91.00%. Finally, the CapsNet model reaches the accuracy of 96.00%. In this comparative analysis, the CapsNet model reached better accuracy and other performance than the other two classifier models. Table 4 and Figure 6 present multi-class classification.

Table 4 represents the comparative analysis of multi-class classification of different models such as RNN, recurrent neural network, and the pretrained model of CNN (CapsNet). The classifier model of the recursive neural network reached a sensitivity of 93.00% and an accuracy of 89.00%. The recurrent neural network model reaches an accuracy of 84.00%. Finally, the CapsNet model reaches an accuracy of 93.00%. In this comparative analysis, the CapsNet model reached better accuracy and other performance than the other two classifier models.

Table 5 and Figure 7 present the comparative analysis of various pretrained models of CNN in terms of accuracy for binary classification, and the proposed model shows better accuracy in binary classification than multi-class classification, which is shown in Figure 8.

Table 5 represents the comparative analysis of accuracy evaluation for binary classification using different classifier models such as UNet, ResNet, VGG-16, EfficientNet,

Table 2: Comparative analysis for segmentation [26].

Method	DSC	JSI	AC
FCM	82.1	66.6	84.8
FRFCM	91.1	72.4	92.6
ARKFCM	92.0	85.8	96.5
Proposed MTDL	93.5	87.8	97.1

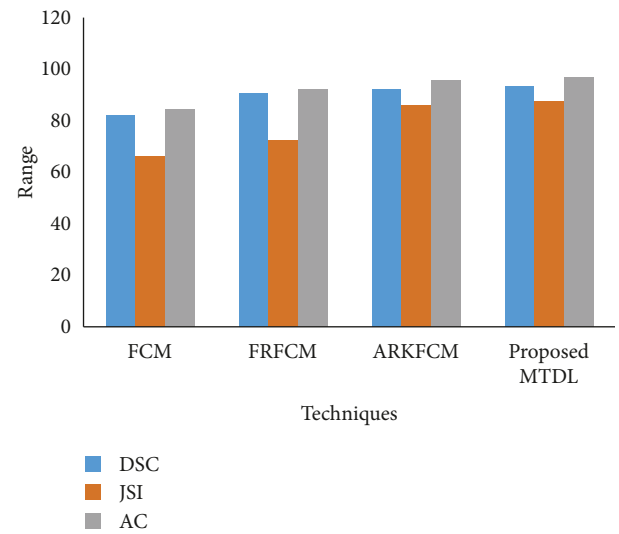


Figure 4: Graphical comparison of proposed segmentation with existing FCM models.

Table 3: Binary classification for the proposed classifier with existing techniques.

Classifier model with DHO	SE	SP	AC
Recursive neural network (RNN)	93.00	96.00	90.00
Recurrent neural network	93.00	96.00	91.00
Pretrained model of CNN (CapsNet)	97.00	98.00	96.00

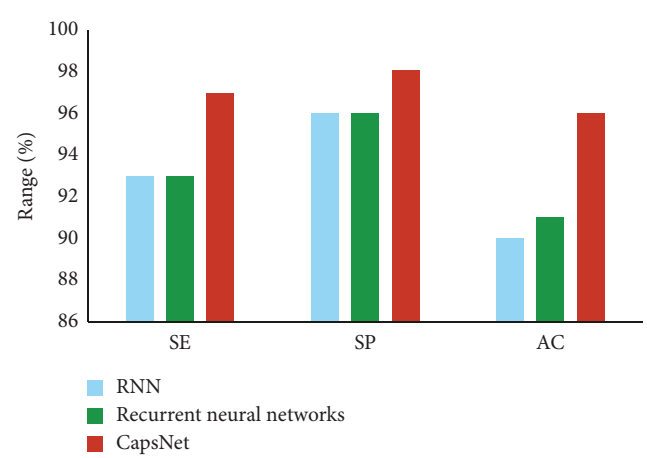


Figure 5: Graphical comparison of proposed classifier with existing models.

Table 4: Multi-class classification for the proposed classifier with existing techniques.

Classifier model with DHO	SE	SP	AC
Recursive neural network (RNN)	89.00	90.00	88.00
Recurrent neural network	87.00	90.00	84.00
Pretrained model of CNN (CapsNet)	91.00	90.00	93.00

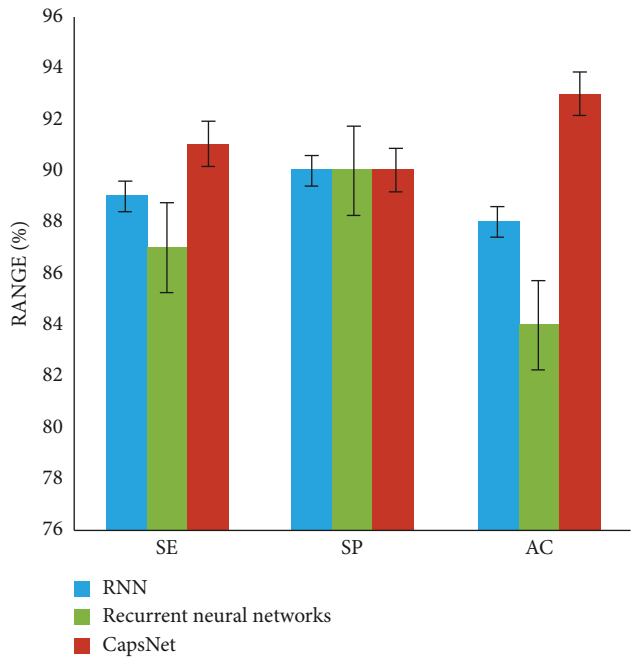


Figure 6: Graphical comparison of proposed classifier with existing models for multi-class classification.

Table 5: Accuracy evaluation for binary classification.

Classification network	Accuracy
UNet	89.00
ResNet	87.00
VGG-16	81.11
EfficientNet	71.11
CapsNet	90.00
UNet-DHO	94.00
ResNet-DHO	95.33
VGG-16-DHO	95.50
EfficientNet-DHO	93.50
CapsNet-DHO	**96.00**

Bold value shows the highest accuracy.

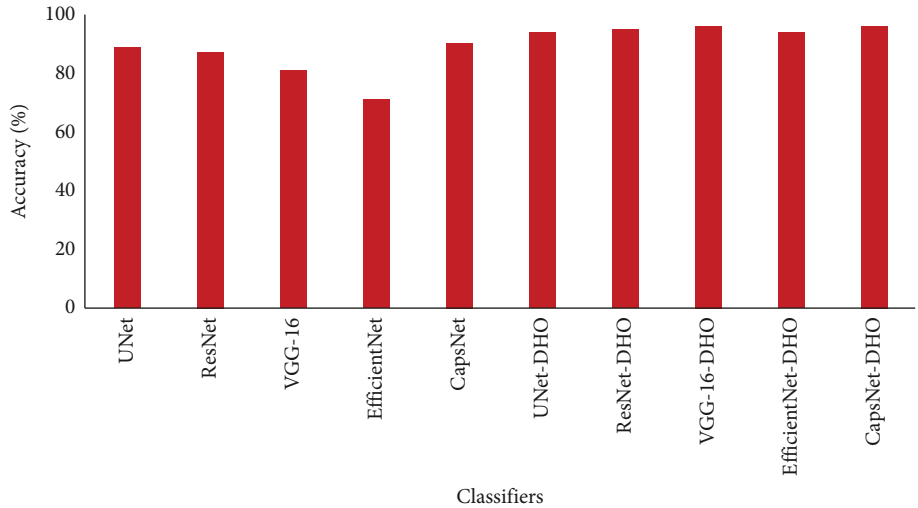


Figure 7: Accuracy comparison for the proposed pretrained model.

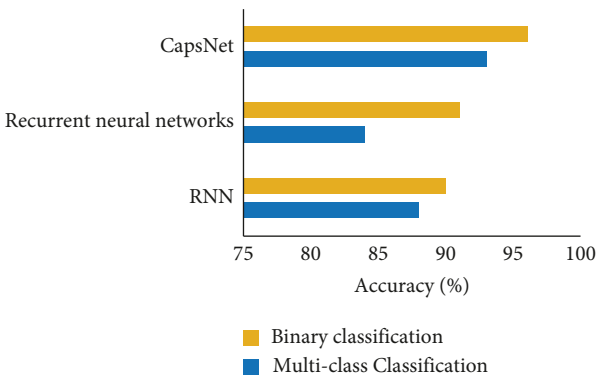


Figure 8: Comparison of the proposed model with existing techniques in terms of accuracy for binary and multi-class classification.

CapsNet, UNet-DHO, ResNet-DHO, VGG-16-DHO, EfficientNet-DHO, and CapsNet-DHO models. Initially, the UNet model reached an accuracy of 89.00%. The ResNet model reached an accuracy of 87.00%. The CapsNet model reached an accuracy of 90.00%. But, when pretrained models are implemented with DHO, it starts to show better accuracy. VGG-16-DHO model reached an accuracy of 95.50%.

Finally, the CapsNet-DHO model reaches the accuracy of 96.00%. By this comparative analysis, the CapsNet-DHO model reached a better accuracy than other classifier models.

5. Conclusion

This research study successfully developed and analyzed the MRI data using a deep learning framework for combined automatic hippocampus segmentation and AD categorization. Multi-task deep learning (MTDL) is used to learn hippocampus segmentation simultaneously. The hyperparameter optimization of the CNN model (capsule network) for illness classification is then carried out using the deer hunting optimization (DHO) technique. ADNI-standardized MRI datasets have been used to test the suggested method, and it is accurate. Suggested MTDL achieved 97.1% accuracy and 93.5% of Dice coefficient, whereas the proposed MTDL model achieved an accuracy of 96% for binary classification and 93% for multi-class classification. Also, in accuracy evaluation for binary classification, the CapsNet-DHO reached a better accuracy performance than other classifier models. The proposed MTDL reached a better accuracy of 97.1% and achieved better performance than other methods. In the analysis of DSC, FCM achieved 82.1%, FRFCM achieved 91%, ARKFCM achieved 92%, and the proposed model achieved 93.5%. Finally, JSI is high for MTDL (i.e., 87.8%) compared to existing FCM models (69% to 85%). The model considered only one dataset for validation, and as a future work, real-time data will be collected and used for verification process. In addition, the efficiency of the pretrained model of CNN will be validated, where the hybrid DL model will be designed for identification of real-time collected AD images [27].

Data Availability

The data used to support the findings of the study are included within the article and are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] A. Emadi, M. Ozen, and A. Abdi, "A hybrid model to study how late long-term potentiation is affected by faulty molecules in an intraneuronal signaling network regulating transcription factor CREB," *Integrative Biology*, vol. 14, pp. 111–125, 2022.

[2] A. Association, "2018 Alzheimer's disease facts and figures," *Alzheimer's and Dementia*, vol. 14, pp. 367–429, 2018.

[3] B. De strooper and E. Karran, "The cellular phase of Alzheimer's disease," *Cell*, vol. 164, pp. 603–615, 2016.

[4] D. P. Veitch, M. W. Weiner, P. S. Aisen, L. A. Beckett, N. J. Cairns, and R. C. Green, "Understanding disease progression and improving Alzheimer's disease clinical trials: recent highlights from the Alzheimer's disease neuroimaging initiative," *Alzheimers Dement*, vol. 15, pp. 106–152, 2019.

[5] S. Rathore, M. Habes, M. A. Iftikhar, A. Shacklett, and C. Davatzikos, "A review on neuroimaging-based classification studies and associated feature extraction methods for Alzheimer's disease and its prodromal stages," *NeuroImage*, vol. 155, pp. 530–548, 2017.

[6] D. Lu and Q. Weng, "A survey of image classification methods and techniques for improving classification performance," *International Journal of Remote Sensing*, vol. 28, pp. 823–870, 2007.

[7] J. Samper-Gonzalez, N. Burgos, S. Bottani, S. Fontanella, P. Lu, and A. Marcoux, "Reproducible evaluation of classification methods in Alzheimer's disease: framework and application to MRI and PET data," *NeuroImage*, vol. 183, pp. 504–521, 2018.

[8] B. C. Riedel, M. Daianu, G. VerSteeg, A. Mezher, L. E. Salminen, and A. Galstyan, "Uncovering biologically coherent peripheral signatures of health and risk for Alzheimer's disease in the aging brain," *Frontiers in Aging Neuroscience*, vol. 10, p. 390, 2018.

[9] S. M. Plis, D. R. Hjelm, R. Salakhutdinov, E. A. Allen, H. J. Bockholt, and J. D. Long, "Deep learning for neuroimaging: a validation study," *Frontiers in Neuroscience*, vol. 8, p. 229, 2014.

[10] Y. Lecun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, p. 436, 2015.

[11] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: a review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 1798–1828, 2013.

[12] C. Farabet, C. Couprie, L. Najman, and Y. Lecun, "Learning hierarchical features for scene labeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, pp. 1915–1929, 2013.

[13] M. Hutson, "Artificial intelligence faces reproducibility crisis," *Science*, vol. 359, pp. 725-726, 2018.

[14] F. U. R. Faisal and G. R. Kwon, "Automated detection of alzheimer's disease and mild cognitive impairment using whole brain MRI," *IEEE Access*, vol. 1921, 2022.

[15] J. V. Shanmugam, B. Duraisamy, B. C. Simon, and P. Bhaskaran, "Alzheimer's disease classification using pretrained deep networks," *Biomedical Signal Processing and Control*, vol. 71, Article ID 103217, 2022.

[16] L. F. Samhan, A. H. Alfarra, and S. S. Abu-Naser, "Classification of Alzheimer's disease using convolutional neural networks," *International Journal of Academic Information Systems Research (IJAISR)*, vol. 6, no. 3, 2022.

[17] J. Tian, G. Smith, H. Guo et al., "Modular machine learning for Alzheimer's disease classification from retinal vasculature," *Scientific Reports*, vol. 11, no. 1, pp. 1–11, 2021.

[18] R. Divya and R. ShanthaSelvaKumari, "Genetic algorithm with logistic regression feature selection for Alzheimer's disease classification," *Neural Computing & Applications*, vol. 33, no. 14, pp. 8435–8444, 2021.

[19] N. An, H. Ding, J. Yang, R. Au, and T. F. Ang, "Deep ensemble learning for Alzheimer's disease classification," *Journal of Biomedical Informatics*, vol. 105, Article ID 103411, 2020.

[20] J. Zhang, B. Zheng, A. Gao, X. Feng, D. Liang, and X. Long, "A 3D densely connected convolution neural network with connection-wise attention mechanism for Alzheimer's disease classification," *Magnetic Resonance Imaging*, vol. 78, pp. 119–126, 2021.

[21] C. R. Jack, M. A. Bernstein, N. C. Fox et al., "The Alzheimer's disease neuroimaging initiative (ADNI): MRI methods," *Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine*, vol. 27, no. 4, pp. 685–691, 2008.

[22] E. T. Tan, L. Marinelli, Z. W. Slavens, K. F. King, and C. J. Hardy, "Improved correction for gradient nonlinearity effects in diffusion-weighted imaging," *Journal of Magnetic Resonance Imaging*, vol. 38, no. 2, pp. 448–453, 2013.

[23] J. G. Sled, A. P. Zijdenbos, and A. C. Evans, "A nonparametric method for automatic correction of intensity nonuniformity in MRI data," *IEEE Transactions on Medical Imaging*, vol. 17, no. 1, pp. 87–97, 1998.

[24] L. Khedher, J. Ramírez, J. M. Górriz, A. Brahim, and F. Segovia, "Alzheimer's Disease Neuroimaging Initiative, et al., Early diagnosis of Alzheimer's disease based on partial least squares, principal component analysis and support vector machine using segmented MRI images," *Neurocomputing*, vol. 151, pp. 139–150, 2015.

[25] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," 2017, https://arxiv.org/abs/1710.09829.

[26] P. Priyadharshini and B. S. E. Zoraida, "Computed tomography imaging lung cancer segmentation using TGVFCM and classification using deep learning techniques," *Journal of Green Engineering*, vol. 10, no. 10, pp. 9184–9203, 2020.

[27] J. E. Galvin, "Prevention of Alzheimer's disease: lessons learned and applied," *Journal of the American Geriatrics Society*, vol. 65, pp. 2128–2133, 2017.