Swarm Optimised Few-View Binary Tomography

Mohammad Majid al-Rifaie*[1][0000−0002−1798−9615] and Tim Blackwell[2][0000−0002−2886−8931]

[1] Univeristy of Greenwich, London SE10 9LS, UK
m.alrifaie@gre.ac.uk
[2] Goldsmiths, University of London, London SE14 6NW, UK
t.blackwell@gold.ac.uk

Abstract. This paper considers a swarm optimisation approach to few-view tomographic reconstruction. DFOMAX, a high diversity swarm optimiser, demonstrably reconstructs binary images to a high fidelity, outperforming a leading algebraic technique, differential evolution and particle swarm optimisation on four standard phantoms. The paper considers the effectiveness of optimisers that have been developed for optimal low dimensional performance and concludes that trial solution clamping on the walls of the feasible search space is important for good performance.

Keywords: Swarm optimisation · Binary tomography · High dimensional optimisation

1 Introduction

Tomographic reconstruction (TR), which is the determination of the internal structure of an opaque object from projected images cast by penetrating radiation, is at the heart of all medical imaging and has widespread application, for example: data compression and data security [24], image processing [35], electron microscopy [14], crystal structure [9], angiography [15], nondestructive testing of homogeneous objects [22], seismic tomography [30], astronomy [12] and geometric, combinatorial and recreational mathematics [19].

Projection data is typically noisy due to the inherent randomness of radiation, detector characteristics and, in medical applications, patient movement, and is usually too sparse for a complete reconstruction. The number of projections should be kept to an absolute minimum in medical contexts due to the damaging effects of radiation. The *few-view* situation is particularly important where the risk is too high, for example in imaging of children.

The standard tomographic numerical reconstruction technique has been, until recently, filtered backprojection (FBP). This algorithm only requires a single iteration but it depends on a large number of projections and is not suitable for few-view imaging [21]. Algebraic Reconstruction Techniques (ART) [24] have effectively replaced FBP in the last few years. ART is an iterative algorithms

* Corresponding author

based on Kaczmarz's method for solving linear system of equations [35]. ART is applicable to the few-view scenario but can introduce artefacts due to overfitting and has not been proven in large patient populations [20].

Compressed sensing (CS) holds potential for few-view imaging: exact reconstruction is possible if the data can be transformed to a sparse representation [36]. However the sparse representation must be known, and the method involves replacing a non-convex problem with a tractable convex minimisation [13]. Iterative statistical methods have also been applied to TR. These algorithms maximise the likelihood of parameters of an underlying statistical model. MLEM, and an accelerated version known as OSEM, also suffer from overfitting, but noise amplification can be reduced with MAP regularisation. Deep learning (DL), despite its success in natural language and computer vision applications, is yet to improve upon traditional analytical methods [28].

Optimisation techniques, and in particular population-based algorithms, are complementary to analytical methods and hold promise for few-view problems due to their resilience, relative lack of assumptions and ability to succeed where analytical methods fail. Several metaheuristic algorithms have been applied to TR, including harmony search [32], tabu search [26], simulated annealing [25], memetics [16] and evolutionary algorithms [8]. Swarm algorithms have also been trialled in binary reconstruction [29], geophysical reconstruction [37], electrical capacitance and impedance tomography [38, 23] and surface reconstruction from 3D data [18]. An algorithm based on the movement of particles over a single image, a pixel-swarm, has been developed for binary reconstruction [4].

The few-view TR problem is underdetermined, which indicates the lack of unique solution. An optimisation might find an image of low loss but there is no guarantee that this image is medically feasible. ART and other least-squares methods tend to produce low norm, diffuse solutions with small pixel values. Swarm algorithms make no assumption (such as low norm or sparsity) about the nature of the solution and do not require convexity.

This paper reports on the application of swarm algorithms to few-view binary TR. After an account of binary tomography and swarm algorithms, the feasibility of high dimensional optimisation without any specific coping mechanism such as subspace optimisation dimensional search is considered. Four swarm algorithms are trialled on four standard phantoms and a suggestive mechanism for the effectiveness of wall-clamping is proposed.

2 Binary tomographic reconstruction

Incident radiation is typically modelled by a projection matrix $A \in \mathbb{R}_{\geq 0}^{m \times n}$ where m is the total number of rays (projections) and n is the number of pixels in the reconstructed image. Suppose that $b \in \mathbb{R}^m$ is a vector of detector values. Then the continuous/discrete reconstruction problem can be stated as:

$$\text{find } x \begin{cases} \in \mathbb{R}^n \\ \in \{0, 1, \ldots, k-1\}^n, k > 1 \end{cases} \quad \text{such that } Ax = b.$$

In the binary problem, $k = 2$ i.e. $x \in \{0, 1\}^n$.

The equation $Ax = b$ cannot be inverted if $m < n$: an approximate solution y must be found. This trial solution is forward projected:

$$Ay = c$$

with a *reconstruction* error

$$e_1(y) = ||b - c||_1 \tag{1}$$

An iterative scheme will produce a sequence of candidate solutions of decreasing error but, due to underdetermination, low reconstruction error does not imply faithfulness to the original object x^*. The proximity of y to x^* can be measured:

$$e_2 = ||y - x^*||_1 \tag{2}$$

In cases where x^* is known, this *reproduction* error provides a test of the ability of an algorithm to find a faithful reconstruction.

3 Swarm optimisation

An optimisation swarm, for real-space problems $\arg\min f(x)$, where $f : X \to \mathbb{R}$ and $X \subset \mathbb{R}^n$ is the feasible search space, is a population of interacting 'particles'. Each particle position is a possible solution; particles move under each others' influence in an attempt to improve the best found position. Particle interactions might be mediated by current or historical positions of particles in a spatial or social neighbourhood. Two swarm optimisers and differential evolution, a real-space population algorithm that has much in common with swarms, are described below.

PSO In particle swarm optimisation, particles i in a canonical PSO swarm [34, 27, 33] of M particles are a triple (x_i, v_i, p_i), representing position, velocity and personal best (pbest), p_i, of the best position they have achieved in the run, as measured by the objective function f. Dynamical variables are updated by the rule

$$\begin{aligned} v_i(t+1) &= wv_i(t) + cu_1 \circ (n_i(t+1) - x_i(t)) \\ &+ \; cu_2 \circ (p_i(t+1) - x_i(t)) \\ x_i(t+1) &= x_i(t) + v_i(t+1) \end{aligned} \tag{3}$$

where $u_{1,2} \sim U(0, 1)$ are uniform random variables in $[0, 1]^D$ and $\circ$ is the Hadamard (entry-wise) product, n_i is the pbest of the best neighbour in i's social network (an arbitrary choice is made in the case of a tie). The inertial weight, w, and acceleration coefficients c, are two arbitrary (but constrained) positive real parameters chosen to balance convergence and exploration and t labels iteration. The pbests may be determined synchronously at the start of the iteration or asynchronously on a particle-by-particle basis.

Two social networks are common in PSO implementations: a global network where particles have access to all pbests (GPSO) and a local ring (LPSO) network where particles can only access 'left' and 'right' neighbours. LPSO has slower information transport; this property inhibits convergence and favours early exploration. LPSO is generally better at more complex multi-modal problems [10].

DE Differential evolution has many variants. We specify the DE/best/1 version, which is considered competitive and robust [17].

Iterations begin with a determination of the current position, g of the best particle. Then, for each particle i, indices j and k are selected such that $i \neq j \neq k$. A random component $r \in \{1, 2 \dots n\}$ is also selected. Component d of particle i at x_i is updated:

$$\begin{aligned} &\text{if } u \sim U(0,1) < C_R \text{ or } d == r \\ &\quad y_d = g_d + F(x_{jd} - x_{kd}) \\ &\text{else} \\ &\quad y_d = x_{id} \end{aligned} \tag{4}$$

where y is a trial position and the parameters $C_R \in [0, 1]$ and $F \in [0, 2]$ are known as the 'cross-over rate' and the 'differential weight'. Then, after each component of y has been set, i is conditionally moved:

$$x(t+1) = \arg\min^*(f(y), f(x(t)))$$

DFO Dispersive flies optimisation [1], is a slim PSO variant without memory and velocity whose exploration and exploitation behaviour is studied in [2]. Updates are based on instantaneous, rather than historical, position. In addition, it incorporates component-wise particle jumps [11]. The best overall position $g(t+1)$ and best ring neighbours $n_i(t+1)$ are determined (with arbitrary choices in the case of ties). Component d of all particles i other than the swarm best (written x_{id}) updates according to

$$\begin{aligned} &\text{if } u \sim U(0,1) < \Delta \\ &\quad x_{id}(t+1) \sim U(X_d) \\ &\text{else} \\ &\quad x_{id}(t+1) = n_{id}(t+1) + \phi u_1(g_d(t+1) - x_{id}(t)) \end{aligned} \tag{5}$$

where Δ is a preset jump probability, $U(X_d)$ is the uniform distribution along axis d of the search space X, $u_1 \sim U(0, 1)$ and $\phi \in [0, \sqrt{3}]$. ϕ is invariably set to 1 and Δ to 0.001 in published studies (e.g. [3, 6, 31, 7, 5]). The upper bound on ϕ is derived from a convergence analysis for stochastic difference equations [11]). DFOMAX will henceforth denote DFO with $\phi = \sqrt{3}$. DFOMAX is expected to have the maximum diversity since the particle update rule places the swarm on the edge of divergence.

DFO uses both global and local strategies and formally, with its reliance on instantaneous position, abandonment of particle memory and retention of

a static communication network, interpolates between PSO and DE. Formal comparison does not of course necessarily imply intermediate performance.

4 Constrained search in high dimensions

The TR problem is high dimensional: a modest 32×32 image has 1024 pixels, therefore, the search space has 1024 dimensions. There is a question whether the algorithms that have been developed for low dimensions (typically $n = 30$), such as the three algorithms specified above, will adapt to the high dimensional problem. There are further issues regarding boundaries and placement of global optima.

The feasible search space for image reconstruction is $X = [0, 255]^n$. A particle might fly outside X and the algorithm must specify if any action is to be taken. Particles may continue to move in $\overline{X} = \mathbb{R}^n \setminus X$ and are either evaluated (if f is defined in $\overline{X}$) or remain unevaluated. Alternatively, particles might be clamped to the boundary ∂X i.e. $x_{id} = \max(\min(255, x_{id}), 0)$. Furthermore, algorithms might behave differently for problems with the global minimum in the interior of X or on ∂X.

The five swarm algorithms defined above (DE, DFO, DFOMAX, GPSO and LPSO) were tested on the Sphere problem, $f(x) = x \cdot x$, with $X = [0, 255]^n$ and with the global optimum at 128^n (interior) and at 255^n (boundary) in a range of dimensions, n, and for clamped and freely moving particles (which are not evaluated outside X. This problem, despite its unimodality and symmetry is far from trivial in high dimensions and with optimum on bounds.

A swarm size of $M = 100$ was chosen for DE, DFO, DFOMAX and G/LPSO. Particles were initialised in X with the uniform distribution and G/LPSO velocities were set to zero. The DFO jump probability Δ was set to 0.001; G/LPSO was run with $w = 0.729844$ and $c = 1.49618$ and the DE/best/1 parameters F and C_R were both set to 0.5.

Tables 1 - 4 report on median errors after 30 runs of 10^5 function evaluations for each algorithm under different boundary conditions and placement of global optimum.

Table 1 shows median errors for the sphere function with optimum in the centre of X and no boundary action where particles can move freely and are evaluated everywhere. All algorithms struggle in higher dimensions where the swarms have barely improved upon their initialised best value: $\mathbb{E}f(x) = n \int_0^{255} x^2 \frac{dx}{255} = 21675n \approx 2 \times 10^7$ $(n = 1000)$. DFOMAX is very poor in lower dimensions, presumably because the ϕ-parameter promotes a very high diversity. Placing the optimum at the corner of X (Table 2) does not change the picture; errors are even higher in higher dimensions indicating that this is a harder problem for free movement boundary conditions.

Tables 3 and 4 show the corresponding experiments with clamped swarms i.e. particles straying outside X are immediately projected onto ∂X. Clamping has no effect when the optimum is in the middle of the search space, but drastically improves DFO, DFOMAX and LPSO performance for high dimensions when

Table 1: Sphere: Particles move freely and are evaluated everywhere, $x^* = 128^n$

Dimension	DE	DFO	DFOmax	GPSO	LPSO
50	1.18e-17	7.53e-12	8.65e+02	3.63e-09	1.40e+00
100	2.26e-05	5.14e-04	3.46e+04	3.04e-01	1.83e+03
200	1.19e+02	1.21e+01	2.91e+05	4.87e+03	9.03e+04
300	5.76e+03	5.80e+02	6.62e+05	5.15e+04	4.70e+05
400	3.51e+04	4.53e+03	1.12e+06	2.41e+05	1.09e+06
500	8.78e+04	1.79e+04	1.54e+06	1.62e+06	1.63e+06
600	1.62e+05	4.52e+04	2.12e+06	2.08e+06	2.14e+06
700	2.65e+05	9.31e+04	2.61e+06	2.48e+06	2.53e+06
800	3.64e+05	1.64e+05	3.15e+06	2.94e+06	3.02e+06
900	5.01e+05	2.58e+05	3.71e+06	3.43e+06	3.45e+06
1000	6.47e+05	3.80e+05	4.27e+06	3.84e+06	3.83e+06

Table 2: Sphere: Particles move freely and are evaluated everywhere, $x^* = 255^n$

Dimension	DE	DFO	DFOMAX	GPSO	LPSO
50	1.81e-17	3.43e-11	1.67e+03	6.67e-09	7.13e+00
100	1.21e-04	2.25e-03	1.19e+05	1.47e+00	8.72e+03
200	8.03e+02	5.81e+01	1.08e+06	3.82e+04	4.10e+05
300	6.13e+04	2.59e+03	2.54e+06	2.95e+05	1.55e+06
400	2.53e+05	2.19e+04	4.21e+06	8.70e+05	3.18e+06
500	6.58e+05	8.36e+04	5.94e+06	1.67e+06	5.04e+06
600	1.17e+06	2.17e+05	7.68e+06	2.65e+06	7.28e+06
700	1.88e+06	4.38e+05	9.78e+06	3.70e+06	9.45e+06
800	2.69e+06	7.65e+05	1.17e+07	5.00e+06	1.21e+07
900	3.53e+06	1.18e+06	1.37e+07	6.54e+06	1.49e+07
1000	4.52e+06	1.70e+06	1.59e+07	8.63e+06	1.70e+07

the optimum is placed at the corner of X. Setting ϕ to its maximum value is advantageous at $n = 1000$; in this case the larger diversity is aiding search.

The Sphere problem offers some clues to the relationship between large n performance, optimum placement and boundary conditions but is different in several important respects from the reconstruction problem $f_{TR}(x) = ||b - Ax||_1$.

f_{TR} has an infinity of solutions lying on the $(n - m)$ dimensional hyperplane $\{x : Ax = b\}$, a rather extreme multimodality. A found exact solution might not equate to the original image, x^*, i.e. zero reconstruction error (e_1) does not ensure zero reproduction error (e_2). Level sets of f_{TR} are flat in $n - m$ dimensions; the Sphere's level sets are curved in n dimensions. High-n curvature renders update unlikely because the interior of the level set has a much smaller volume than the exterior. However level set flatness will remain problematic in few-view TR because the topography remains curved in $m \approx n$-dimensions.

A dummy TR problem was devised in order to test if the Sphere results might generalise. The five algorithms were trialled on a uniform phantom $x^* = 255^{32\times32}$ under free movement and clamping. The solution hyperplane intersects with

Table 3: Sphere: Particles are clamped to X= $[0, 255]^D$, $x^* = 128^n$

Dimension	DE	DFO	DFOMAX	GPSO	LPSO
50	6.29e-18	9.37e-12	1.01e+02	2.59e-09	9.09e-01
100	2.01e-05	4.54e-04	9.40e+03	1.64e+04	1.00e+03
200	1.55e+02	1.21e+01	1.43e+05	7.05e+04	3.72e+04
300	1.01e+04	5.32e+02	4.41e+05	1.89e+05	1.57e+05
400	5.22e+04	4.40e+03	8.40e+05	3.37e+05	3.53e+05
500	1.31e+05	1.66e+04	1.28e+06	5.31e+05	5.81e+05
600	2.33e+05	4.38e+04	1.77e+06	7.66e+05	8.31e+05
700	3.49e+05	8.89e+04	2.27e+06	1.01e+06	1.09e+06
800	5.40e+05	1.52e+05	2.77e+06	1.29e+06	1.36e+06
900	6.51e+05	2.37e+05	3.28e+06	1.54e+06	1.65e+06
1000	8.53e+05	3.49e+05	3.82e+06	1.87e+06	1.90e+06

Table 4: Sphere: Particles are clamped to X= $[0, 255]^D$, $x^* = 255^n$

Dimension	DE	DFO	DFOMAX	GPSO	LPSO
50	3.23e-27	0.00e+00	0.00e+00	0.00e+00	0.00e+00
100	4.80e-25	0.00e+00	0.00e+00	2.60e+05	0.00e+00
200	3.64e-11	0.00e+00	0.00e+00	9.75e+05	0.00e+00
300	4.96e-06	0.00e+00	0.00e+00	1.85e+06	0.00e+00
400	2.68e-03	0.00e+00	0.00e+00	2.96e+06	0.00e+00
500	2.53e-01	0.00e+00	0.00e+00	4.00e+06	0.00e+00
600	4.34e+00	0.00e+00	0.00e+00	4.97e+06	0.00e+00
700	3.26e+04	0.00e+00	0.00e+00	6.08e+06	0.00e+00
800	6.53e+04	0.00e+00	0.00e+00	7.51e+06	0.00e+00
900	1.31e+05	2.29e-08	0.00e+00	8.49e+06	0.00e+00
1000	2.00e+05	5.69e+01	0.00e+00	9.88e+06	0.00e+00

$X = [0, 255]^{1024}$ at a single point (a corner) so the problem is unimodal in X. Tables 5 and 6 report on $e_{1,2}$ under the above experiment settings and for $m = 6, 8, 16, 32$. Free movement performance is poor for all algorithms and for all numbers of projections m. The reproduction error even exceeds the maximum value of $261120 = 32 \times 32 \times 255$ in some instances, indicating that best positions lie outside X. Clamping tells a different story: both errors are improved and DFO and DFOMAX find the corner optimum corresponding to the original phantom. We see that clamping can play an important role in TR, rendering even high-n problems tractable.

5 Reconstructions

Four standard binary TR test phantoms, as depicted in the leftmost column of Fig. 1, were chosen for the algorithm trials. Two sizes, 32×32 and 64×64 and 6, 8, 16 and 32 projections (few-view scenarios) were tested. Phantom imaging was

Table 5: Uniform 32×32 phantom: particles move freely and are evaluated everywhere

e_1	DE	DFO	DFOMAX	GPSO	LPSO
$m = 6$	27512	36946	229072	61075	194348
$m = 8$	52050	62997	331839	107265	285517
$m = 16$	188214	181379	781586	348052	674266
$m = 32$	472357	381192	1618504	847332	1444595

e_2	DE	DFO	DFOMAX	GPSO	LPSO
$m = 6$	287601	423121	529438	139996	150539
$m = 8$	281366	388361	509529	141577	150056
$m = 16$	245387	318266	472701	137075	153196
$m = 32$	211071	228996	433595	144072	151037

Table 6: Uniform 32×32 phantom, clamped particles

e_1	DE	DFO	DFOMAX	GPSO	LPSO
$m = 6$	25000	0	0	434558	30689
$m = 8$	33865	0	0	586305	39757
$m = 16$	73463	0	0	1163191	83098
$m = 32$	137865	0	0	2316471	176054

e_2	DE	DFO	DFOMAX	GPSO	LPSO
$m = 6$	4612	0	0	77392	5865
$m = 8$	4737	0	0	77775	5610
$m = 16$	5006	0	0	77902	5865
$m = 32$	4804	0	0	77647	6120

conducted by the ASTRA tomography toolbox[3] using parallel geometry with the number of rays set to 32 and 64 for the the 32×32 and 64×64 phantoms respectively.

Six algorithms were chosen, including the five swarm (DE is considered here as a pseudo-swarm) algorithms from the previous section and SIRT, an algebraic reconstruction algorithm from the toolbox. A previous examination (paper under review) with the same experiment set-up confirmed that SIRT was the superior algebraic algorithm. In addition, random search (RS) was used as a control because the swarm algorithms rely on extensive sampling. All data from DE, DFO, G/LPSO, SIRT and RS experiments are taken from a recent paper (under review).

Particles were clamped to $[0, 255]$ in each dimension and swarms and RS were run for 100,000 function evaluations. Swarm parameter settings were identical to those of Section 4 experiments. All algorithms with randomisation were run 30 times on each problem.

[3] https://www.astra-toolbox.com

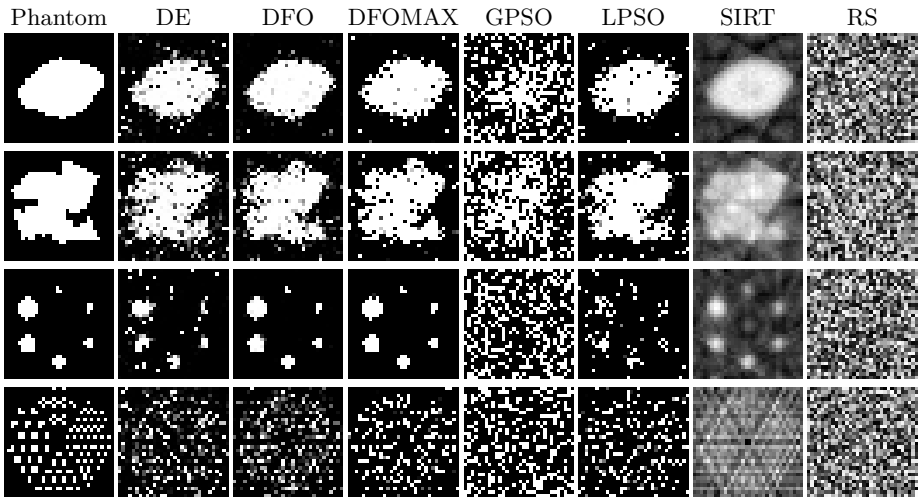

Fig. 1: Phantoms $1-4$ (top to bottom) and sample reconstructions by the algorithms, with phantom size of 32×32 and 6 projections.

6 Results

Fig. 2 depicts algorithm reconstructions for each of the eight problems. Reconstruction improves in all cases with increased projections and for the smaller phantom. The extreme few-view case ($64 \times 64, m = 6$) is very challenging. SIRT produces blurred images and swarm reconstructions are sharper but contain pepper noise. GPSO is particularly bad, with reconstructions that are hardly better than random search. Otherwise, DFO and DFOMAX appear to be the best performers.

Table 7 confirms these visual findings. DFO and DFOMAX find exact reproductions for phantoms 1, 2 and 3 of size $n = 32^2$ at 32 projections and phantoms 1 and 3 at $m = 16$. DFOMAX recovers the 32^2 phantom 3 at all projections, and in all the runs. Reconstruction of the 64^2 is more difficult because of the extreme few-view scenario.

Wilcoxon rank tests of algorithm performance (e_1 and e_2) were conducted at $\alpha = 0.05$. The results of the 32 problems is reported in Table 8. SIRT dominates the rankings for e_1 but produces significantly worse reproductions (e_2) than all algorithms apart from GPSO and RS. DFOMAX is the top reproduction performer, beating DE, GPSO and SIRT in each problem; DFOMAX also returned better reconstruction errors (e_1) than DE and GPSO. Setting DFO's ϕ to $\sqrt{3}$ and thereby ensuring maximum diversity without explosion improves DFO in this setting (beating DFO 25 times and 26 times for reconstruction and reproduction error respectively).

The results of the preliminary trials on optimum-on-corner-Sphere and the uniform phantom are confirmed for these phantoms. It seems that clamping

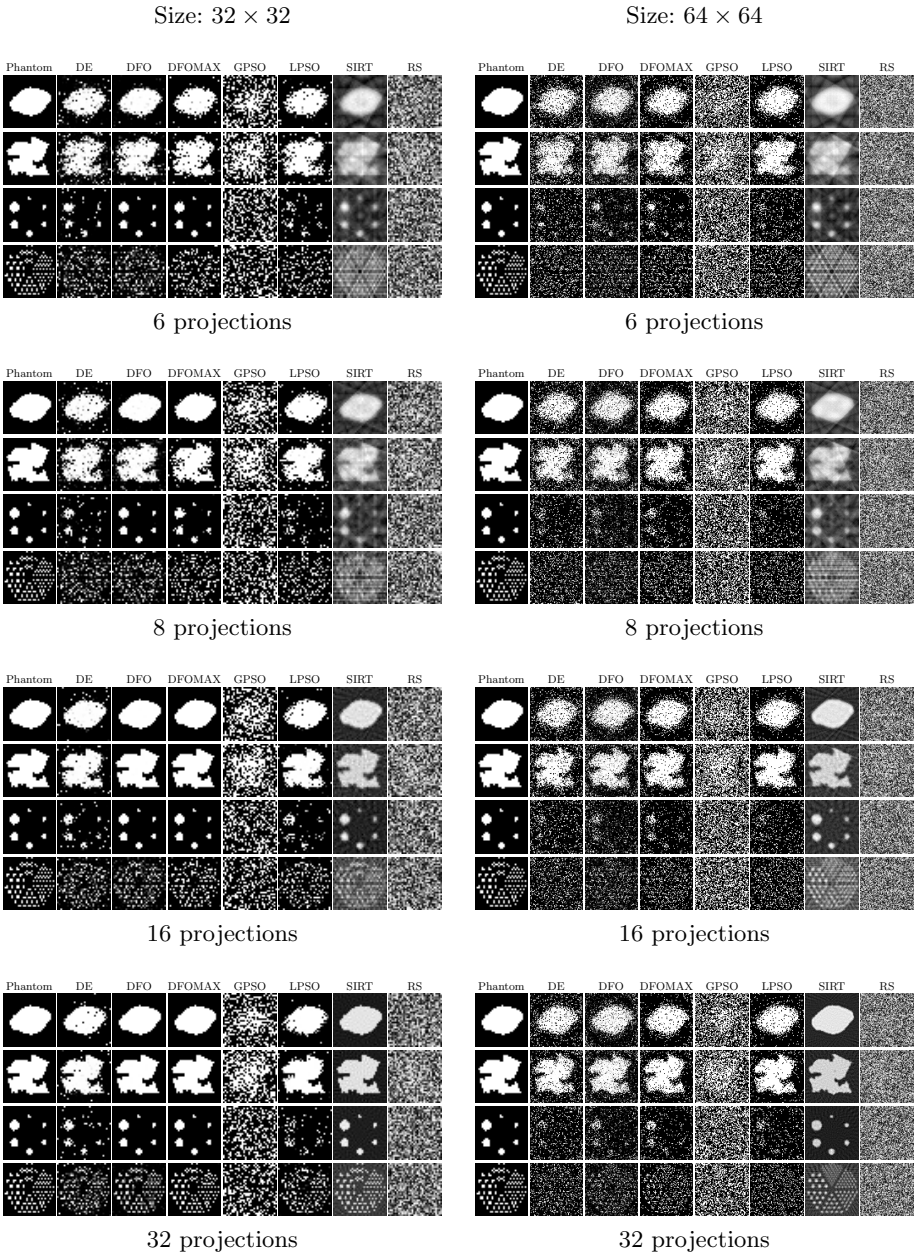

Fig. 2: Reconstructed phantoms

Table 7: Rounded median reproduction error, e_2, for each problem and each algorithm. Lighter shading indicates the proximity of the reconstructions to the phantoms. The largest error in phantoms of sizes 32^2 and 64^2 are 255×32^2 and 255×64^2 respectively.

	DE	DFO	DFOMAX	GPSO	LPSO	SIRT	RS
Phantom 1, size = 32^2, $m = 6$	24360	11392	9462	90896	17006	52254	123172
$m = 8$	18306	4250	0	88231	12444	53452	122473
$m = 16$	10800	0	0	87185	8898	42305	123530
$m = 32$	8646	0	0	87993	8540	27998	122754
Size = 64^2, $m = 6$	210940	207107	169545	435764	154123	246054	506638
$m = 8$	193099	191829	151688	432352	137348	241197	507789
$m = 16$	172757	171379	130625	430594	117964	179971	507951
$m = 32$	163515	160018	117159	430219	109999	150775	507091
Phantom 2, size = 32^2, $m = 6$	39667	27905	27878	93150	28901	75948	122240
$m = 8$	35209	22588	22280	92225	22997	71291	122429
$m = 16$	17269	785	0	90415	8816	52369	122056
$m = 32$	11701	0	0	87261	7921	30212	122628
Size = 64^2, $m = 6$	251333	243877	210000	438343	190386	308809	507088
$m = 8$	237651	230159	196678	436555	170015	280526	506604
$m = 16$	189399	188739	151242	428432	120831	241244	504912
$m = 32$	171271	171116	128279	430698	109442	181168	504911
Phantom 3, size = 32^2, $m = 6$	14213	697	0	84787	17259	61049	122377
$m = 8$	17187	2302	0	88102	20912	65759	121866
$m = 16$	12000	0	0	86317	18689	40385	122053
$m = 32$	10536	0	0	85807	18903	31715	121947
Size = 64^2, $m = 6$	173616	161874	126078	425212	156488	243334	504370
$m = 8$	181518	166219	134297	423937	159573	297473	504085
$m = 16$	178099	158374	121136	421260	160890	219134	505120
$m = 32$	175900	155551	117687	424702	149612	141821	504338
Phantom 4, size = 32^2, $m = 6$	54862	48549	48357	93915	52286	106043	123660
$m = 8$	58459	51857	54111	94987	56592	101792	122876
$m = 16$	48886	37014	37679	95115	44677	90408	123858
$m = 32$	40083	17595	11636	94222	38095	60032	123279
Size = 64^2, $m = 6$	251265	231391	226598	437839	244786	369251	472486
$m = 8$	261236	242350	244098	443354	253149	402474	471607
$m = 16$	252098	225371	222921	442305	244071	378158	473273
$m = 32$	242980	212200	202150	440890	231698	250803	473312

Table 8: Algorithms comparison based on e_1 and e_2. The numbers indicate statistically significant wins for the algorithm in the left hand column versus the algorithm in the top row. Total number of problems are 32: 4 phantoms (1,2,3,4) $\times$ 2 sizes (32^2, 64^2) $\times$ 4 projections (6,8,16,32)

e_1	DE	DFO	DFOMAX	GPSO	LPSO	SIRT	RS
DE	NA	0	0	32	4	0	32
DFO	32	NA	6	32	23	5	32
DFOMAX	32	25	NA	32	24	9	32
GPSO	0	0	0	NA	0	0	32
LPSO	24	8	7	32	NA	0	32
SIRT	32	27	23	32	32	NA	32
RS	0	0	0	0	0	0	NA

e_2	DE	DFO	DFOMAX	GPSO	LPSO	SIRT	RS
DE	NA	0	0	32	4	30	32
DFO	29	NA	1	32	18	30	32
DFOMAX	32	26	NA	32	22	32	32
GPSO	0	0	0	NA	0	2	32
LPSO	27	11	8	32	NA	31	32
SIRT	2	2	0	30	1	NA	32
RS	0	0	0	0	0	0	NA

ameliorates the curse of high dimension in cases where the desired optimum is on the boundary of the search space. The results for DFO and DFOMAX are very promising considering the few-view conditions and the high dimensionality of the problems. The apparent feasibility of a conversion of reconstruction into an optimisation problem is surprising where the swarm is moving in the space of all possible 32^2 or 64^2 binary images.

7 Discussion

The results of all trials indicate the importance of imposing clamping in cases where the desired optimum lies on a corner of the search space. This finding can be supported by considering a model algorithm optimising Sphere.

The model algorithm is assumed to produce trial points with spherical symmetry. In particular, trials are generated in a ball of radius r with probability $\frac{1}{2}$ about a centre x. For the Sphere problem, x lies on a hyperspherical level set of radius R. All points within the level set have lower function value; a trial point is accepted if it lies in the n-ball of radius R which we can assume is centred at O. If P_u is the relative volume of intersection of the trial ball $B_n(x, r)$ with the ball of points with lower function value $B_n(O, R)$, i.e.

$$P_u = \frac{\mathrm{vol}(B_n(x, r) \cap B_n(O, R))}{\mathrm{vol}(B_n(x, r))}$$

then the probability of updating in a single trial is $\frac{P_u}{2}$. Fig. 3 illustrates three possibilities for three different search radii. The geometry is depicted in Fig. 4.

The update region $B_n(x,r) \cap B_n(O,R)$ in the leftmost diagram of Fig. 3 is an n-dimensional lens. The length of the major axis, a, of this lens is $R \sin\phi$ (Fig. 4). a is larger than the minor axes for $\phi \in (0,\pi)$. From Fig. 4, $\sin\phi = \rho\sqrt{1-\frac{\rho^2}{4}}$, and, with $\rho = \frac{r}{R}$,

$$P_u < \frac{(R\sin\phi)^n}{r^n} = \left(1-\frac{\rho^2}{4}\right)^{\frac{n}{2}} \quad (\rho < 1)$$

For $\rho \geq 1$, middle and rightmost diagrams of Fig. 3, $B_n(x,r) \cap B_n(O,R)$ is covered by a ball of radius R, so

$$P_u \leq \rho^{-n} \quad (\rho \geq 1)$$

Hence $\lim_{n\to\infty} P_u = 0$ for all scenarios in which $\rho > 0$. The pathology of high dimensions is manifest as a varnishing update probability for spherically symmetric search on the Sphere function; for example, in 1000 dimensions and $\rho = \frac{1}{2}$, the chances of finding a better trial position are less than 10^{-14}.

The above conclusions is valid for unconstrained search. Clamping at a wall significantly increases the update probability. Fig. 5 depicts wall clamping for optimisation of the Sphere with optimum at a corner. A trial landing in region A will be projected onto the edge of the search box and into the update region. Furthermore, a particle that has been repositioned on the wall will have a significantly greater chance of updating (because region A is a half-space) at the next iteration. The effective augmentation of the update region by clamping is a possible mechanism for the evident improved performance of all swarm algorithms for corner problems (although it does not explain the superiority of this pair over PSO and DE).

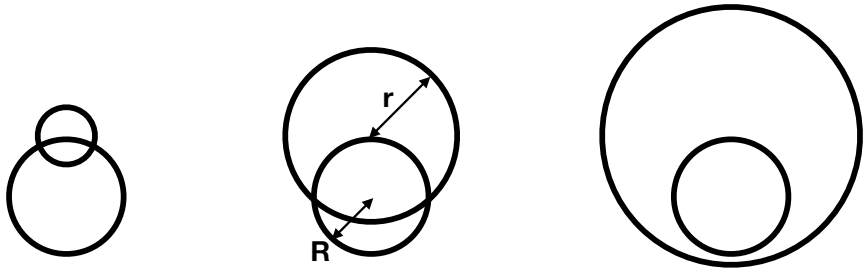

Fig. 3: Three configurations of the update region $B_n(x,r) \cap B_n(O,R)$

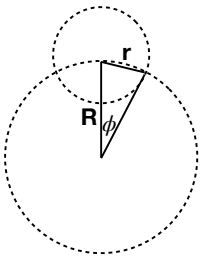

Fig. 4: Geometry of the lens-shaped update region

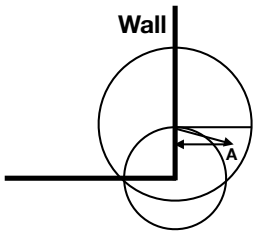

Fig. 5: Advantageous effect of clamping

8 Conclusions

This paper establishes that swarm search algorithms developed for optimisation in low dimensions ($n \sim 30$) can transfer to reconstruction problems in higher dimension ($n = 1024-4096$) providing that particles are clamped to the walls of the search space. Experiments with four standard phantoms under few-view conditions demonstrate good reconstructions when compared to the original phantom for two swarm optimisers: DFO and DFOMAX. The latter is a diversity boosted version of the former and offers superior performance. The higher diversity possibly encourages motion outside the search space; particles are clamped back to the search space wall, effectively increasing update probability. Theoretical arguments based on an idealised optimiser support this picture.

DFOMAX reconstructions are comparable to SIRT, the best algebraic technique for the problems under consideration. DFOMAX can achieve sharper and higher fidelity reconstructions in the few-view regime. Further tuning of DFO to this problem set (swarm size and jump probability remain fixed at optimal settings for low dimensional problems) is important in the light of these findings. An extension of these trials to discrete tomography is also of interest.

8.1 Acknowledgement

The authors would like to thank Darren Wise for his support in facilitating access to the HPC machines at the University of Greenwich.

References

1. al-Rifaie, M.M.: Dispersive flies optimisation. In: M. Ganzha, L. Maciaszek, M.P. (ed.) Proceedings of the 2014 Federated Conference on Computer Science and Information Systems. Annals of Computer Science and Information Systems, vol. 2, pp. pages 529–538. IEEE (2014). https://doi.org/10.15439/2014F142
2. al-Rifaie, M.M.: Investigating knowledge-based exploration-exploitation balance in a minimalist swarm optimiser. In: IEEE Congress on Evolutionary Computation, 2021. CEC'21. IEEE (2021)

3. al-Rifaie, M.M., Aber, A.: Dispersive flies optimisation and medical imaging. In: Recent Advances in Computational Optimization, pp. 183–203. Springer (2016)
4. al-Rifaie, M.M., Blackwell, T.: Binary tomography reconstruction by particle aggregation. In: European Conference on the Applications of Evolutionary Computation. pp. 754–769. Springer (2016)
5. al-Rifaie, M.M., Cavazza, M.: Evolutionary optimisation of beer organoleptic properties: A simulation framework. Foods **11**(3), 351 (Jan 2022). https://doi.org/10.3390/foods11030351
6. al-Rifaie, M.M., Ursyn, A., Zimmer, R., Javid, M.A.J.: On symmetry, aesthetics and quantifying symmetrical complexity. In: Correia, J., Ciesielski, V., Liapis, A. (eds.) Computational Intelligence in Music, Sound, Art and Design: EvoMUSART 2017, pp. 17–32. Springer International Publishing (2017). https://doi.org/10.1007/978-3-319-55750-2_2
7. Aparajeya, P., Leymarie, F.F., al-Rifaie, M.M.: Swarm-based identification of animation key points from 2d-medialness maps. In: Ekárt, A., Liapis, A., Castro Pena, M.L. (eds.) Computational Intelligence in Music, Sound, Art and Design. pp. 69–83. Springer International Publishing, Cham (2019)
8. Batenburg, K.J., Kosters, W.A.: Solving nonograms by combining relaxations. Pattern Recognition **42**(8), 1672–1683 (2009)
9. Batenburg, K.J., Palenstijn, W.J.: On the reconstruction of crystals through discrete tomography. In: International Workshop on Combinatorial Image Analysis. pp. 23–37. Springer (2004)
10. Blackwell, T., Kennedy, J.: Impact of communication topology in particle swarm optimization. IEEE Transactions on Evolutionary Computation **23**(4), 689–702 (2019)
11. Blackwell, T.: A study of collapse in bare bones particle swarm optimization. IEEE Transactions on Evolutionary Computation **16**(3), 354–372 (2011)
12. Butala, M., Hewett, R., Frazin, R., Kamalabadi, F.: Dynamic three-dimensional tomography of the solar corona. Solar Physics **262**(2), 495–509 (2010)
13. Candes, E.J., Romberg, J.K., Tao, T.: Stable signal recovery from incomplete and inaccurate measurements. Communications on pure and applied mathematics **59**(8), 1207–1223 (2006)
14. Carazo, J.M., Sorzano, C., Rietzel, E., Schröder, R., Marabini, R.: Discrete tomography in electron microscopy. In: Discrete Tomography, pp. 405–416. Springer (1999)
15. Carvalho, B., Herman, G., Matej, S., Salzberg, C., Vardi, E.: Binary tomography for triplane cardiography. In: Information Processing in Medical Imaging. pp. 29–41. Springer (1999)
16. Cipolla, M., Bosco, G.L., Millonzi, F., Valenti, C.: An island strategy for memetic discrete tomography reconstruction. Information Sciences **257**, 357–368 (2014)
17. Das, S., Suganthan, P.N.: Differential evolution: A survey of the state-of-the-art. IEEE Transactions on Evolutionary Computation **15**(1), 4–31 (2011). https://doi.org/10.1109/TEVC.2010.2059031
18. Gálvez, A., Iglesias, A.: Particle swarm optimization for non-uniform rational b-spline surface reconstruction from clouds of 3d data points. Information Sciences **192**, 174–192 (2012)
19. Gardner, R.J.: Geometric tomography, vol. 1. Cambridge University Press Cambridge (1995)
20. Geyer, L.L., Schoepf, U.J., Meinel, F.G., Nance Jr, J.W., Bastarrika, G., Leipsic, J.A., Paul, N.S., Rengo, M., Laghi, A., De Cecco, C.N.: State of the art: iterative ct reconstruction techniques. Radiology **276**(2), 339–357 (2015)

21. Giussani, A., Hoeschen, C.: Imaging in nuclear medicine. Springer (2013)
22. Hampel, U., Bieberle, A., Hoppe, D., Kronenberg, J., Schleicher, E., Sühnel, T., Zimmermann, F., Zippe, C.: High resolution gamma ray tomography scanner for flow measurement and non-destructive testing applications. Review of scientific instruments **78**(10), 103704 (2007)
23. Hu, G., Chen, M.y., He, W., Zhai, J.q.: Clustering-based particle swarm optimization for electrical impedance imaging. Advances in Swarm Intelligence pp. 165–171 (2011)
24. Irving, R., Jerrum, M.: Three-dimensional data security problems. SIAM J. Comput. **23**, 170–184 (1994)
25. Jarray, F., Tlig, G., Dakhli, A.: Reconstructing hv-convex images by tabu research approach. In: International Conference on Metaheuristics and Nature Inspired Computing. p. 3 (2010)
26. Jarray, F., Tlig, G.: A simulated annealing for reconstructing hv-convex binary matrices. Electronic Notes in Discrete Mathematics **36**, 447–454 (2010)
27. Kennedy, J.: Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance. In: In: Proceedings of the 1999, Congress of Evolutionary Computation. vol. 3, pp. 1931–1938. IEEE Press (1999)
28. Lucas, A., Iliadis, M., Molina, R., Katsaggelos, A.K.: Using deep neural networks for inverse problems in imaging: Beyond analytical methods. IEEE Signal Processing Magazine **35**(1), 20–36 (2018)
29. Miklós, P.: Particle swarm optimization approach to discrete tomography reconstruction problems of binary matrices. In: Intelligent Systems and Informatics (SISY), 2014 IEEE 12th International Symposium on. pp. 321–324. IEEE (2014)
30. Nolet, G., et al.: A breviary of seismic tomography. Imaging the Interior (2008)
31. Oroojeni, H., al-Rifaie, M.M., Nicolaou, M.A.: Deep neuroevolution: Training deep neural networks for false alarm detection in intensive care units. In: European Association for Signal Processing (EUSIPCO) 2018. pp. 1157–1161. IEEE (2018). https://doi.org/10.23919/EUSIPCO.2018.8552944
32. Ouaddah, A., Boughaci, D.: Improving reconstructed images using hybridization between local search and harmony search meta-heuristics. In: Proceedings of the Companion Publication of the 2014 Annual Conference on Genetic and Evolutionary Computation. pp. 1475–1476. ACM (2014)
33. Poli, R., Kennedy, J., Blackwell, T.: Particle swarm optimization: An overview. Swarm Intellligence **1**, 33–57 (2007)
34. Shi, Y., Eberhart, R.: A modified particle swarm optimizer. In: Congress on Evolutionary Computation. pp. 69–73 (1998)
35. Shliferstein, A.R., Chien, Y.: Some properties of image-processing operations on projection sets obtained from digital pictures. IEEE Transactions on Computers **26**(10), 958–970 (1977)
36. Tao, T.: Compressed sensing or: the equation ax= b, revisited. Mahler Lecture Series (2009)
37. Tronicke, J., Paasche, H., Böniger, U.: Crosshole traveltime tomography using particle swarm optimization: A near-surface field example. Geophysics **77**(1), R19–R32 (2012)
38. Wang, P., Lin, J., Wang, M.: An image reconstruction algorithm for electrical capacitance tomography based on simulated annealing particle swarm optimization. Journal of applied research and technology **13**(2), 197–204 (2015)