Invited Paper

Bifurcation analysis by particle swarm optimization

Haruna Matsushita\textsuperscript{1a)}, Hiroaki Kurokawa\textsuperscript{2}, and Takuji Kousaka\textsuperscript{3}

\textsuperscript{1} Dept. of Electronics and Information Engineering, Kagawa University, 2217–20 Hayashi-cho, Takamatsu, Kagawa 761–0396, Japan
\textsuperscript{2} Dept. of Electric and Electronic Engineering, Tokyo University of Technology, 1404–1 Katakuramachi, Hachioji City, Tokyo 192–0982, Japan
\textsuperscript{3} Dept. of Electrical and Electronic Engineering, Chukyo University, 101–2 Yagoto Honmachi, Showa-ku, Nagoya-shi, 466–8666, Japan

\textsuperscript{a)} haruna@ieee.org

Received January 31, 2020; Revised May 13, 2020; Published October 1, 2020

Abstract: This paper explains a bifurcation parameter detection strategy based on particle swarm optimization (PSO), and it shows application examples on detection of local bifurcation parameters appearing in discrete-time dynamical systems and non-autonomous continuous dynamical systems. The algorithm design and analysis part shows that the PSO-based algorithm is fairly simple, easily understandable, and easily implementable. The method requires no careful initialization, exact calculation, gradient information of the system, or Lyapunov exponents. However, the simulation results show that the method accurately detects the local bifurcation parameters regardless of the stability of the periodic point.

Key Words: bifurcation point detection, bifurcation diagram, initial setup problem, particle swarm optimization (PSO)

1. Introduction

Complex real-world dynamics or phenomena depend on time and can be modeled as dynamical systems. A system described by ordinary differential equations (ODEs) is said to be continuous dynamical systems, and a system described by difference equations is said to be a discrete-time dynamical system or a map. Furthermore, by using switching, and/or combining differential equations and maps, various real-world systems can be described by mathematical formulas as the dynamical systems. Thus, analyzing such dynamical systems is important to solve a variety of problems in the real world, and dynamical systems research covers a wide range, embracing engineering, biology, sociology, ecology, and other fields. Such dynamical systems mostly contain one or more parameters. In general, the behavior is nonlinear and bifurcation phenomena can be observed. The bifurcation occurs when a small change, applied to the parameter values of a system (referring to bifurcation parameter), causes a sudden qualitative change in the periodic solutions’ behavior. Bifurcation analysis, which investigates
how bifurcations depend on the system parameters, is one of the most important nonlinear analysis

techniques for understanding phenomena of systems since small changes in the parameter values may

have large effects on the system behaviors.

A bifurcation diagram, which is a graph of bifurcation sets, is often used to visualize a bifurcation

structure. The simplest method for providing a two-parameter bifurcation diagram is a brute-force

method. However, since an accuracy of the brute-force bifurcation diagram depends on the number

of points on the grid in target parameter space, higher accuracy bifurcation diagram requires larger

amount of calculations. Furthermore, due to its character, the brute-force method cannot detect

unstable periodic point which is dynamically unstable. Meanwhile, various detection and tracing

methods have been developed for bifurcation parameters [1–4], and a number of software packages

are available for bifurcation analysis, such as AUTO [5] (and its specialized drivers such as SLIDE-

CONT [6]) and BunKi [7]. However, these standard methods are based on the Newton–Raphson

method, which is a gradient-based algorithm that requires the derivative of a function and appro-

priate initial conditions. Thus, such methods can be difficult for beginners who are unfamiliar with

nonlinear analysis.

Alternatively, bifurcation analysis can be performed by population-based optimization (also called

swarm intelligence) which is a metaheuristic algorithm with multiple potential solutions. Most

population-based optimizations avoid the derivative of the objective functions, and a particularly pop-

ular algorithm is Particle Swarm Optimization (PSO) [8]. Despite its simplicity, the PSO performs

global search and has many real/potential applications. Our research group proposed a PSO-based

method to detect bifurcation parameters in discrete-time and non-autonomous dynamical systems [9–

11]. The method, called nested layer particle swarm optimization (NLPSO), is a non-Newton method

and is performed by two nested PSOs. Although the NLPSO requires no strict settings of initial

values or cumbersome manual calculations, it can accurately and directly find the bifurcation pa-

rameter set without a Lyapunov or gradient-based method, regardless of the periodic point stability.

The initial value problem of Newton-based bifurcation analyses can then be solved by assigning the

NLPSO-detected single bifurcation parameter set as the initial point of the bifurcation curve tracing.

Furthermore, NLPSO can also obtain a bifurcation diagram of both unstable and stable periodic

points by plotting multiple bifurcation parameter sets detected by repeating the implementation of

the NLPSO with different random initial states.

This paper explains the NLPSO-based detection strategy of the bifurcation parameter sets ap-

pearing in discrete-time dynamical systems and non-autonomous continuous dynamical systems. The

paper is written with the needs of an expert at the dynamical systems in mind, however, those less

familiar with the bifurcation analysis or the swarm intelligence should also be able to understand it.

The rest of this paper is organized as follows. Section 2 provides the PSO algorithm. Section 3

defines dynamical systems and the bifurcation of its periodic solutions. Section 4 describes the struc-

ture and algorithm of the NLPSO by incorporating two objective functions. Section 5 establishes

the validity and applicability of NLPSO-based method using several examples. Section 6 provides

conclusions and identifies several directions for future research.

2. Particle swarm optimization (PSO)

The PSO is one of the simplest and most popular population-based optimization techniques that

exploits a population of individuals that correspond to multiple potential solutions. The population is
called swarm, and the individuals (i.e., the objective variables of the optimization) are called particles.

PSO searches the optimal solution by moving the particles around in the search space according to

simple updating equations. Each particle moves with being influenced by its personal best position

and the swarm best position. This has an effect of converging the swarm to a better solution.

PSO is a metaheuristic optimization method since it makes few or no assumptions about the

optimization problem being solved. However, PSO does not guarantee that a global optimum can be

found, and the solution found depends on the set of random variables generated because it includes

some stochastic factors. Meanwhile, although PSO avoids the derivative of the optimization problem
Each particle are initialized to zero to prevent particles from leaving the search space during the first iterations.

D

Suppose a initial conditions for the optimization.

spaces of potential solutions. These characteristics mean that PSO is applicable to discontinuous

Newton method and gradient descent, and it can find good solutions by searching over very large

(i.e. the objective function of the optimization) as is required by gradient-based algorithms such as

Newton method and gradient descent, and it can find good solutions by searching over very large

spaces of potential solutions. These characteristics mean that PSO is applicable to discontinuous

objective function with switch or jump, and it can find the optimal solution without appropriate initial

conditions for the optimization.

The PSO algorithm is explained as follows. Each particle carries position and velocity information.

Suppose a D-dimensional search space \( S \) and a swarm contains \( M \) particles, the position and velocity vectors of the \( i \)-th particle at iteration \( h \) are represented by \( z_i(h) = (z_{i1}, z_{i2}, \ldots, z_{iD}) \in \mathbb{R}^D \) and \( v_i(h) = (v_{i1}, v_{i2}, \ldots, v_{iD}) \in \mathbb{R}^D \), where \( i = 1, 2, \ldots, M \). Table I summarizes the symbols used in PSO.

Initially, at \( h = 0 \), the PSO particles are randomly distributed in the search space \([z_{\text{min}}, z_{\text{max}}]^D\), where \( z_{\text{min}} \) and \( z_{\text{max}} \) represent the lower and upper boundaries of the search-space \( S \). The velocities are initialized to zero to prevent particles from leaving the search space during the first iterations. Each particle \( i \) moves toward its personal best position \( p_i(h) = (p_{i1}, p_{i2}, \ldots, p_{iD}) \) \((pbest)\), which defines their best previous position and the global best position \( g(h) = (g_1, g_2, \ldots, g_D) \) \((gbest)\), that defines the best \( pbest \) with the best objective value among all particles, i.e., \( g(h) = p_g(h) \), where \( g \) is a \( gbest \) index defined by

\[
g = \arg \min_i F(p_i(h)),
\]

where \( F \) is the objective function under consideration. Thus, \( g(h) \) is the global optimum solution at the iteration \( h \). The elements of \( v_i \) and \( z_i \) in each dimension \( d \) of a particle \( i \) are updated as follows [12]:

\[
\begin{align*}
v_{id}(h + 1) &= w v_{id}(h) + r_1 c_1 (p_{id}(h) - z_{id}(h)) + r_2 c_2 (g_d(h) - z_{id}(h)), \\
z_{id}(h + 1) &= z_{id}(h) + v_{id}(h + 1),
\end{align*}
\]

where \( d = 1, 2, \ldots, D \), and \( r_1 \) and \( r_2 \) are two random numbers uniformly distributed in the interval \([0,1]\), taking a different value in each dimension \( d \). The inertia weight \( w \) determines how much of the particle’s previous velocity is preserved, and \( c_1 \) and \( c_2 \) are fixed positive acceleration coefficients.

This study set these three parameters to their optimal default values [13], namely, \( w = 0.729 \) and \( c_1 = c_2 = 1.494 \). These processes are repeated until the user-defined stop criterion is reached. Thus, \( g(h_{\text{end}}) \) is the solution obtained by PSO, where \( h_{\text{end}} \) is the iteration counts at which the stop criterion is satisfied.

Particles sometimes can be attracted to regions outside the feasible search space \( S \). One of the methods to preserve the feasibility of the found solution is in which particles located outside the

---

**Table I.** Variables and their symbols of two PSOs in NLPSO.

| Variable             | Symbol | PSO | PSO\(_{bif}\) | PSO\(_{pp}\) |
|----------------------|--------|-----|---------------|-------------|
| Particle index       |        | \( i \) |               |             |
| Index of \( gbest \) |        | \( g \) |               |             |
| Dimension index      |        | \( d \) |               |             |
| Iteration counter    |        | \( h \) | \( h_1 \) | \( h_2 \) |
| Position vector      |        | \( z_i \) | \( z_{bi} \) | \( z_{pi} \) |
| Velocity vector      |        | \( v_i \) | \( v_{bi} \) | \( v_{pi} \) |
| \( pbest \) vector   |        | \( p_i \) | \( p_{bi} \) | \( p_{pi} \) |
| \( gbest \) vector   |        | \( g \) | \( g_b \) | \( g_p \) |
| Number of particles  |        | \( M \) | \( M_{bif} \) | \( M_{pp} \) |
| Number of search space dimensions | | \( D \) | \( L \) | \( N \) |
| Objective function   |        | \( F(\cdot) \) | \( F_{bif}(\cdot) \) | \( F_{pp}(\cdot) \) |
| Maximum iterations   |        | \( H \) | \( H_1 \) | \( H_2 \) |
| Stop criteria        |        | \( C \) | \( C_{bif} \) | \( C_{pp} \) |

---

393
Algorithm 1 Pseudo code of the PSO.

▷ Objective variable $z$
▷ Objective function $F(z)$.

Randomly initialize the position $z$ of each particle $i$ by uniformly distributing the particles over the search space $[z_{\text{min}}, z_{\text{max}}]^D$.

Initialize other particle variables as $v_i(0) = 0$, $p_i(0) = z_i(0)$, $F(p_i(0)) = \infty$ and $F(g(0)) = \infty$.

Iteration counter $h \leftarrow 0$.

while $(h < H \AND F(g(h)) > C)$ do
  
  $h \leftarrow h + 1$

  for $i = 1$ to $M$, all $M$ particles do
    
    Calculate $F(z_i(h))$.

    if $F(z_i(h)) < F(p_i(h-1))$ then
      
      Update $p_{\text{best}}$ of particle $i$ as $p_i(h) = z_i(h)$.
    
    else
      
      $p_i(h) = p_i(h-1)$.
    
    end if

  end for $i$

  Update $g_{\text{best}}$ as $g(h) = p_g(h)$, where $g = \arg\min_i F(p_i(h))$.

  for $i = 1$ to $M$, all $M$ particles do
    
    for $d = 1$ to $D$, all $D$ dimensions do
      
      Update $v_{id}$ and $z_{id}$ according to Eq. (2).
    
    end for $d$

  end for $i$

end while

Output the final results $g(h_{\text{end}})$.

search space cannot become its $p_{\text{best}}$ even though its function value is better than that of previous $p_{\text{best}}$. Therefore, they are attracted back to the feasible search space in subsequent iterations.

Algorithm 1 is the pseudo code of the PSO that minimizes the optimization, where $H$ is the maximum iteration and $C$ is the stop criterion. Thus, PSO terminates the search process when $h > H$ or $F(g(h)) < C$ is satisfied.

3. Definition of dynamical systems

3.1 Discrete-time dynamical system

A general form of the $N$-dimensional discrete-time dynamical system is described as

$$x(k+1) = f(x(k), \lambda),$$

where $k$ denotes discrete time, and $x(k) \in \mathbb{R}^N$ and $\lambda \in \mathbb{R}^L$ correspond to the state variables and system parameters, respectively.

Let $f^l$ denote the $l$-th iteration of $f$. A point $x_p \in \mathbb{R}^N$ is said to be an $n$-periodic point of $f$ if $x_p = f^n(x_p, \lambda)$ and $x_p \neq f^l(x_p, \lambda)$ for $l < n$. A 1-periodic point is referred to as a fixed point.

The Jacobian matrix of $f$ at the $n$-periodic point $x_p$ is described by

$$Df^n(x_p, \lambda) = \prod_{l=0}^{n-1} \frac{\partial}{\partial x(n - 1 - l)} \left( f(x(n - 1 - l), \lambda) \right),$$

with characteristic equation

$$\det (Df^n(x_p, \lambda) - \mu I_N) = 0,$$

where $x(0) = x_p$, $I_N$ denotes the $N \times N$ identity matrix, and $\mu$ is a characteristic multiplier of $Df^n(x_p, \lambda)$. The characteristic multipliers determine the local stability class of an $n$-periodic point. If all characteristic multipliers of $Df^n(x_p, \lambda)$ are inside the unit circle in the complex plane, then $x_p$
is a stable periodic point (SPP). If any characteristic multipliers of $Df^n(x_p, \lambda)$ are outside the unit circle, then $x_p$ is an unstable periodic point (UPP). Period-doubling and saddle-node bifurcations occur under the conditions $\mu = -1$ and $\mu = 1$, respectively. Table II summarizes the variables used in the dynamical system definition.

### 3.2 Continuous dynamical system

We consider an $N$-dimensional continuous dynamical system described by ODEs in Eq. (6):

$$\frac{dx}{dt} = f(t, x, \lambda),$$

where $t \in \mathbb{R}$ denotes the time; $x \in \mathbb{R}^N$ corresponds to the state variables and $\lambda \in \mathbb{R}^L$ are the system parameters. Since the state velocity includes the time $t$, Eq. (6) is called a non-autonomous system.

The function $f : \mathbb{R} \times \mathbb{R}^N \times \mathbb{R}^L \rightarrow \mathbb{R}^N$ is continuously differentiable, namely, smooth function (class $C^\infty$). The map $f$ periodically varies according to the time $t$ with period $T$. A solution of an initial state $x(t_0) = x_0 \in \mathbb{R}^N$ ($t_0 = 0$) is defined as

$$x(t) = \varphi(t, x_0, \lambda).$$

Then the Poincaré map $\Phi$ is described as

$$\Phi : \mathbb{R}^N \rightarrow \mathbb{R}^N; \; x_0 \mapsto x_1 = \Phi(x_0) = \varphi(T, x_0, \lambda).$$

Let $\Phi^l$ denote the $l$-th iteration of the map $\Phi$ which is described as

$$\Phi^l : \mathbb{R}^N \rightarrow \mathbb{R}^N; \; x_0 \mapsto x_l = \Phi^l(x_0) = \Phi \circ \Phi \circ \cdots \circ \Phi(x_0) = \varphi(lT, x_0, \lambda).$$

In other words, the Poincaré map converts the continuous system to the discrete-time system by sampling the orbit from the initial state $x_0$ with the sampling interval $T$ as

$$\{x_0, x_1, x_2, \ldots, x_l, \ldots\} = \{x_0, \Phi(x_0), \Phi^2(x_0), \ldots, \Phi^l(x_0), \ldots\}. \tag{10}$$

An $n$-periodic solution of $\Phi$ is defined as follows:

$$x_p = \Phi^n(x_p) = \varphi(nT, x_p, \lambda) \text{ and } x_p \neq \Phi^l(x_p) \text{ for } l < n. \tag{11}$$

The Jacobian matrix of $\Phi^n$ at the $n$-periodic solution $x_p$ is described as

$$D\varphi(nT, x_p, \lambda) = D\Phi^n(x_p) = \frac{\partial \Phi^n(x_p)}{\partial x_p}, \tag{12}$$

with the characteristic equation:

$$\det(D\Phi^n(x_p) - \mu I_N) = 0. \tag{13}$$

By the way, in Eq. (12), $\partial \Phi^n(x_p)/\partial x_p$ is the matrix described by

$$\frac{\partial \Phi^n(x_p)}{\partial x_p} = \begin{pmatrix}
\frac{\partial \varphi_1(nT, x_p, \lambda)}{\partial x_p} & \frac{\partial \varphi_1(nT, x_p, \lambda)}{\partial x_p} & \cdots & \frac{\partial \varphi_1(nT, x_p, \lambda)}{\partial x_p} \\
\frac{\partial \varphi_2(nT, x_p, \lambda)}{\partial x_p} & \frac{\partial \varphi_2(nT, x_p, \lambda)}{\partial x_p} & \cdots & \frac{\partial \varphi_2(nT, x_p, \lambda)}{\partial x_p} \\
\vdots & \vdots & \ddots & \vdots \\
\frac{\partial \varphi_N(nT, x_p, \lambda)}{\partial x_p} & \frac{\partial \varphi_N(nT, x_p, \lambda)}{\partial x_p} & \cdots & \frac{\partial \varphi_N(nT, x_p, \lambda)}{\partial x_p}
\end{pmatrix} \tag{14}$$

The remaining part of this section explains how to obtain the respective elements in Eq. (14) [4]. By substituting the solution starting from $x_p$ into the system equation Eq. (6), we obtain
Table II. Variables and their symbols of the dynamical system.

| Variable                               | Symbol |
|----------------------------------------|--------|
| Time of the system (for discrete-time systems) | \(k\) |
| Time of the system (for continuous-time systems) | \(t\) |
| State variable vector                   | \(x\) |
| Periodic point vector                   | \(x_p\) |
| System parameter vector                 | \(\lambda\) |
| Dimension of the system                 | \(N\) |
| Number of system parameters             | \(L\) |
| Period on which the system depends      | \(T\) |
| Iteration of the map                    | \(l\) |
| Number of periods of the periodic solution | \(n\) |
| Characteristic multiplier               | \(\mu\) |

\[
\frac{d\varphi(t, x_p, \lambda)}{dt} = f(t, \varphi(t, x_p, \lambda), \lambda) \tag{15}
\]

and

\[
\varphi(0, x_p, \lambda) = x_p. \tag{16}
\]

Partially differentiate both sides of Eq. (15) with respect to \(x_p\):

\[
\frac{\partial}{\partial x_p} \left( \frac{d\varphi(t, x_p, \lambda)}{dt} \right) = \frac{\partial}{\partial x_p} (f(t, \varphi(t, x_p, \lambda), \lambda)). \tag{17}
\]

The order of differentiation on the left-hand side in Eq. (17) is commutative, and then we can obtain

\[
\frac{d}{dt} \left( \frac{\partial \varphi(t, x_p, \lambda)}{\partial x_p} \right) = \frac{\partial f(t, \varphi(t, x_p, \lambda), \lambda)}{\partial x} \bigg|_{x=x_p} \frac{\partial \varphi(t, x_p, \lambda)}{\partial x_p}. \tag{18}
\]

Similarly, from Eq. (16), the initial state

\[
\frac{\partial \varphi(0, x_p, \lambda)}{\partial x_p} = I_N. \tag{19}
\]

Therefore, we can obtain all the elements in Eq. (14) by numerically solving Eq. (18) with initial value described in Eq. (19) from 0 to \(nT\).

4. Bifurcation point detection strategy by using the nested layer PSO (NLPSO)

To detect bifurcation parameters at a periodic point, we must derive not only the system parameters satisfying the bifurcation conditions but also the periodic point depending on the bifurcation parameters. We proposed the NLPSO-based detection strategy [9–11] for detecting local bifurcation points. NLPSO is performed by two PSOs with nested structure to find the system parameters \(\lambda\) that produces a bifurcation phenomenon on an \(n\)-periodic solution \(x_p\) in a dynamical system \(f\). In the NLPSO algorithm, the main PSO (PSO_{bif}) seeks the bifurcation parameter \(\lambda\), and the inner PSO (PSO_{pp}) searches for the periodic point (or periodic solution) \(x_p\) depending on the system parameter information that is given by the position of the PSO_{bif} particles. Although NLPSO requires no carefully set initial system parameters or exact calculation, it can directly and accurately detect the bifurcation parameter without a Lyapunov exponent or a gradient-based method. Please note that our NLPSO-based detection method is for only non-degeneracy bifurcations.

The NLPSO algorithm is here described in detail. Table I summarizes the symbols used in NLPSO. Algorithm 2 is the pseudo code of the PSO_{bif}, which is our proposed NLPSO-based detection strategy for local bifurcation parameters. Algorithm 3 is the pseudo code of the PSO_{pp}, which is invoked...
used in the NLPSO, and we are going to explain this later in detail. Secondly, we set the value of $b_if$ from the PSO Algorithm 3. Pseudo code of PSO

$$\text{Algorithm 2 Pseudo code of the NLPSO for bifurcation parameter detection in a dynamical system.}$$

$\triangleright$ Objective variable $z_b$ corresponding to bifurcation parameter $\lambda$ on the system $f$.

$\triangleright$ Objective function $F_{b_if}(z_b)$.

Iteration counter $h_1 ← 0$.

while $(h_1 < H_1 \text{ AND } F_{b_if}(g_b(h_1)) > C_{b_if})$ do

$h_1 ← h_1 + 1$

for $i = 1$ to $M_{b_if}$, all $M_{b_if}$ particles do

$x^*_p ← \text{PSOPP}(z_b(h_1))$ \hfill $\triangleright$ Carry out the PSO $pp$ in Algorithm 3 with the argument $z_b(h_1)$.

Evaluate $F_{b_if}(z_b(h_1))$ with $x^*_p$ which is a return value of the PSO $pp$.

if $F_{b_if}(z_b(h_1)) < F_{b_if}(p_{b_i}(h_1))$ AND $z_{b_i} \in S_b$ then

Update $p_{b_i}(h_1) = z_{b_i}(h_1)$.

else

Keep $p_{b_i}(h_1) = p_{b_i}(h_1 - 1)$.

end if

end for $i$

Update $\text{gbest } g_b(h_1) = p_{b_{g_b}}(h_1)$, where $g = \arg\min_{i} F_{b_if}(p_{b_i}(h_1))$.

for $i = 1$ to $M_{b_if}$, all $M_{b_if}$ particles do

for $d = 1$ to $L$, all $L$ dimensions do

Update $v_{b_{id}}(h_1)$ and $z_{b_{id}}(h_1)$ according to Eq. (2).

end for $d$

end for $i$

end while

Output the final results $g_b(h_{1\text{end}})$ as a bifurcation parameter $\lambda$.

$\text{Algorithm 3 Pseudo code of PSO}_{pp}$ that finds $n$-periodic point.

procedure PSOPP($\lambda$)

$\triangleright$ Objective variable $z_p$ corresponding to $n$-periodic point $x_p$.

$\triangleright$ Objective function $F_{pp}(z_p)$.

Initialize particle information of the PSO $pp$.

Iteration counter $h_2 ← 0$.

while $(h_2 < H_2 \text{ AND } F_{pp}(g_p(h_2)) > C_{pp})$ do

$h_2 ← h_2 + 1$

for $i = 1$ to $M_{pp}$, all $M_{pp}$ particles do

Evaluate $F_{pp}(z_p(h_2))$.

Update $p_{p_{1i}}(h_2)$ if $F_{pp}(z_p(h_2)) < F_{pp}(p_{p_{1i}}(h_2))$.

end for $i$

Update $\text{gbest } g_p(h_2) = p_{p_{g_p}}(h_2)$, where $g = \arg\min_{i} F_{pp}(p_{p_{i}}(h_2))$.

for $i = 1$ to $M_{pp}$, all $M_{pp}$ particles do

for $d = 1$ to $N$, all $N$ dimensions do

Update $v_{p_{id}}(h_2)$ and $z_{p_{id}}(h_2)$ according to Eq. (2).

end for $d$

end for $i$

end while

return $g_p(h_{2\text{end}})$ \hfill $\triangleright$ The deemed $n$-periodic point depending on $\lambda$ is $g_p(h_{2\text{end}})$.

end procedure

from the PSO $b_if$. We can obtain single bifurcation parameter set by using NLPSO according to the Algorithm 2. Figure 1 shows a flowchart representing a process for detecting multiple different bifurcation parameter sets. Firstly, as preprocessing, we define two objective functions $F_{b_if}$ and $F_{pp}$ used in the NLPSO, and we are going to explain this later in detail. Secondly, we set the value of
μ as −1 or 1, according to the type of target bifurcation that one would like to be found, and set the number of periods \( n \) of the target bifurcation. Next, we initialize PSO\(_{\text{bif}}\) information in the same way to the PSO. After initialization, we carry out the bifurcation parameter search according to the Algorithm 2. The output of the PSO\(_{\text{bif}}\) corresponds to single bifurcation parameter set, and we can obtain multiple bifurcation parameter sets by repeating the implementation of the NLPSO with different random initial states of the particle, according to Fig. 1. In this figure, \( run \) is the simulation counter and \( R_{\text{max}} \) is the maximum number of simulations, thus, the proposed method repeats the bifurcation point search until \( R_{\text{max}} \leq run \) is satisfied.

The remaining part of this section explains two objective functions used in NLPSO in detail. Let us consider a detection of the bifurcation parameters in a discrete-time dynamical system \( f \) defined in Eq. (3). Since the PSO\(_{\text{bif}}\) seeks the bifurcation parameter \( \lambda = (\lambda_1, \lambda_2, \cdots, \lambda_L) \), its particle position \( z_b \) serves the system parameter \( \lambda \). Thus, the PSO\(_{\text{bif}}\) search-space dimension \( D \) is equivalent to the number of the system parameters \( L \) to search, and the \( i \)-th particle position vector at the PSO\(_{\text{bif}}\) iteration count \( h_1 \) is denoted by \( z_b(h_1) = (z_{b1}, z_{b2}, \cdots, z_{bL}) \). Similarly, the \( p\text{best} \) and \( g\text{best} \) are denoted by \( p_b(h_1) = (p_{b1}, p_{b2}, \cdots, p_{bL}) \) and \( g_b(h_1) = (g_{b1}, g_{b2}, \cdots, g_{bL}) \). The objective function \( F_{\text{bif}}(z_b) \) is defined as

\[
F_{\text{bif}}(z_b) = \begin{cases} 
\det \left( Df^n(x_p^*, z_b) - \mu I_N \right), & \text{if } F_{pp}(x_p^*) < C_{pp}, \\
\infty, & \text{otherwise,}
\end{cases}
\]
where $x_p^*$ is a deemed $n$-periodic point detected by the PSO\textsubscript{pp}. $F_{pp}$ is an objective function of the PSO\textsubscript{pp} as will be described later, and $C_{pp}$ is the criterion of the deemed periodic point $x_p^*$.

Equation (20) defines the bifurcation condition at the $n$-periodic point $x_p$ according to Eq. (5), where $F_{bif}(z_b) = 0$ means that $z_b$ produces a bifurcation at $x_p$. Thus, it is a minimization optimization problem with minimum at 0. The PSO\textsubscript{bif} stops searching if the iteration count $h_1$ exceeds the maximum iteration $H_1$ or $gbest$ of the PSO\textsubscript{bif} satisfies $F_{bif}(g_b) < C_{bif}$, where $C_{bif}$ corresponds to the accuracy of the bifurcation point found. Thus, the solution $g_b(h_{1end})$ is the bifurcation parameter detected by NLPSO, where $h_{1end}$ is the iteration counts at which the PSO\textsubscript{bif} satisfied the stop criterion. If we require only the bifurcation parameter located in the target parameter space, the PSO\textsubscript{bif} specifies the feasible search space $S_b$. Note that Eq. (20) guarantees an accuracy of the deemed periodic point $x_p^*$ that composes of the detected bifurcation parameters $\lambda$. When a PSO\textsubscript{bif} particle takes system parameters for which no $n$-periodic solution exists, it does not affect the $gbest$ and $bif$ of the PSO\textsubscript{bif} because its $F_{bif}$ is assigned a bad value ($F_{bif} = \infty$). Therefore, the PSO\textsubscript{bif} particles are attracted to other particles taking system parameters for which $n$-periodic solutions do exist. This effect avoids a false periodic point, that violates $F_{pp}(x_p^*) < C_{pp}$, and is effective for the saddle-node bifurcation parameters detection [10].

To calculate Eq. (20), we must derive the $n$-periodic point $x_p$ depending on the system parameter $\lambda$ corresponding to the PSO\textsubscript{bif} particle position $z_b$. In the NLPSO, the $\lambda$-dependent periodic point is searched by the PSO\textsubscript{pp} nested in the PSO\textsubscript{bif}. Let $M_{bif}$ denote the number of particles in the PSO\textsubscript{bif}. By invoking the PSO\textsubscript{pp} $M_{bif}$ times, using different system parameters given by each $z_b$, the PSO\textsubscript{pp} searches for the $M_{bif}$ periodic solutions, which depend on the respective system parameters. The $i$-th particle position vector at the PSO\textsubscript{pp} iteration count $h_2$ is denoted by $z_{pi}(h_2) = (\tilde{z}_{pi,1}, \tilde{z}_{pi,2}, \ldots, \tilde{z}_{pi,N})$, and it serves the $\lambda$-dependent periodic point $x_p = (x_{p1}, x_{p2}, \ldots, x_{pN})$. Thus, the PSO\textsubscript{pp} search-space dimension $D$ is equivalent to the dimension of the system $N$. Similarly, the $gbest$ and $bif$ are denoted by $p_{gbest}(h_2) = (p_{g1}, p_{g2}, \ldots, p_{gN})$ and $g_p(h_2) = (g_{p1}, g_{p2}, \ldots, g_{pN})$.

The objective function of the PSO\textsubscript{pp} is defined as

$$F_{pp}(z_p) = \| f^n(z_p, \lambda) - z_p \|,$$

where $\| \cdot \|$ denotes the Euclidean distance, and $f^n(z_p, \lambda)$ corresponds to the state variable, which defines the $n$-th iteration of $f$ with initial point $z_p$ and parameter $\lambda$ (see Eq. (3)). Therefore, $F_{pp}(z_p, \lambda)$ is minimized at 0 when $z_p$ is exactly an $n$-periodic point (i.e. $z_p \equiv x_p$). The PSO\textsubscript{pp} stops searching if the iteration count $h_2$ exceeds the maximum iteration $H_2$ or $gbest$ of the PSO\textsubscript{pp} satisfies $F_{pp}(g_p) < C_{pp}$, where $C_{pp}$ corresponds to the accuracy of the periodic point found. Thus, $g_p(h_{2end})$ is the detected solution which is a return value of the PSO\textsubscript{pp}, where $h_{2end}$ is the iteration counts at which the PSO\textsubscript{pp} satisfied the stop criterion. Meanwhile, the $gbest$ $g_p$, detected by the PSO\textsubscript{pp} after searching, is called a deemed $n$-periodic point $x_p^*$ because it is only an estimated periodic point and not guaranteed to be a true $n$-periodic point even if $x_p^*$ satisfied the criterion $F_{pp}(x_p^*) < C_{pp}$. In addition, if a PSO\textsubscript{bif} particle takes system parameters $\lambda$ with no $n$-periodic point, the PSO\textsubscript{pp} can never detect a periodic point. In this case, the PSO\textsubscript{pp} returns a false-periodic point, that violates $F_{pp}(x_p^*) < C_{pp}$, to the PSO\textsubscript{bif} [10].

Here, we make mention of a risk that PSO\textsubscript{pp} finds a non-target periodic point that satisfies $F_{pp}$. If there is a periodic point with a non-target period $k$ in the target parameter space, and $k$ is a divisor of $n$, the non-target periodic point also satisfies Eq. (21). Therefore, on some bifurcation structure, as is the case with avoiding a false-periodic point it is better to give $F_{bif}$ the ability to avoid non-target periodic points that are $k$-periodic points such that $k < n$.

Next, let us consider a bifurcation parameter detection in a non-autonomous system $f$ defined in Eq. (6). Although the basic algorithm is the same as the case in the discrete-time dynamical systems, it can be extended to the non-autonomous system using the Poincaré map. The PSO\textsubscript{bif} objective function $F_{bif}$ to seek a bifurcation parameter in a non-autonomous system is defined as

$$F_{bif}(z_b) = \begin{cases} 
\left| \det (D\Phi^n(x_p^*) - \mu I_N) \right|, & \text{if } F_{pp}(x_p^*) < C_{pp}, \\
\infty, & \text{otherwise,}
\end{cases}$$

(22)
where $\Phi^n (x^*_p)$ is an $n$-periodic solution of a Poincaré map $\Phi$; thus $\Phi^n (x^*_p) = \varphi (nT, x^*_p, z_b)$. Therefore, Eq. (22) is obtained by calculating the Poincaré map of the non-autonomous system.

The PSO$_{pp}$ objective function $F_{pp}$ to seek a $\lambda$-dependent periodic solution $x_p$ in Eq. (22) is defined as

$$F_{pp}(z_p) = \|\Phi^n(z_p) - z_p\|,$$

where $\Phi^n(z_p) = \varphi (nT, z_p, \lambda)$ defines the $n$-th iteration of $\Phi$ with initial point $z_p$ and system parameter $\lambda$ corresponding to a PSO$_{bif}$ particle position $z_b$. This is obtained by considering the Poincaré map in the same way as the PSO$_{bif}$.

5. Simulations

This section demonstrates some examples of 2-parameter bifurcation diagrams obtained by applying the NLPSO to a discrete-time dynamical system and a non-autonomous dynamical system.

5.1 Discrete-time dynamical system — Circle map

The circle map is a one-dimensional map described by

$$x(k + 1) = f(x(k), \lambda) = \left( x(k) + \lambda_1 - \frac{\lambda_2}{2\pi} \sin 2\pi x(k) \right) \mod 1.0,$$

where $x(k)$ is a state variable, and $\lambda = (\lambda_1, \lambda_2)$ are system parameters. The circle map exhibits bifurcation phenomena as these parameters are varied.

From Eq. (20), the objective function of the PSO$_{bif}$, which detects the local bifurcation parameter set, is defined as

$$F_{bif}(z_b) = \begin{cases} \left| \frac{df(x^*_p, z_b)}{dx^*_p} - \mu \right|, & \text{if } F_{pp}(x^*_p) < C_{pp} \\ \infty, & \text{otherwise} \end{cases}$$

$$= \begin{cases} \prod_{l=0}^{n-1} (1 - z_{b2} \cos 2\pi x(n - 1 - l)) - \mu, & \text{if } F_{pp}(x^*_p) < C_{pp} \\ \infty, & \text{otherwise}. \end{cases}$$

where $z_b = (z_{b1}, z_{b2})$ serves $\lambda = (\lambda_1, \lambda_2)$ in Eq. (24); thus, $L = 2$. $x(k)$ is a state variable calculated by Eq. (24), and its initial value $x(0)$ is the deemed $n$-periodic point $x^*_p$ depending on the parameter set $(z_{b1}, z_{b2})$ corresponding to the particle information of the PSO$_{bif}$.

The objective function of the PSO$_{pp}$, which detects $z_b$-dependent deemed periodic point $x^*_p$, is given by

$$F_{pp}(z_p) = |f^n(z_{p1}, \lambda) - z_{p1}|,$$

where $z_p = z_{p1}$ serves $x(0)$ in Eq. (24); thus, $N = 1$. The system parameter vector $\lambda$ in Eq. (26) is an argument of PSO$_{pp}$ and corresponds to a position vector $z_b$ of a single PSO$_{bif}$ particle that invoked the PSO$_{pp}$.

Table III summarizes the experimental parameters of PSO$_{bif}$ and PSO$_{pp}$. The search-space range of the PSO$_{bif}$ is set as

| | PSO$_{bif}$ | PSO$_{pp}$ |
|---|---|---|
| Number of dimensions $D$ | $L$ | $N$ |
| Stop criterion $C_{bif}$, $C_{pp}$ | $1 \times 10^{-3}$ | $1 \times 10^{-5}$ |
| Maximum iterations $H_1$, $H_2$ | 300 | 100 |
| Number of particles $M_{bif}$, $M_{pp}$ | 30 | |
| Inertia weight $w$ | 0.729 | |
| Acceleration coefficients $c_1$, $c_2$ | 1.494 | |
and the initialization range of $z_p$ is set as $[0, 1]$. For different number of periodic points ($n = 2$, $3$ and $5$), NLPSO detected a periodic-doubling bifurcation parameter set ($\mu = -1$) denoted $I^n$ and a saddle-node bifurcation set ($\mu = 1$) denoted $G^n$. These bifurcation parameters were obtained by the same algorithm with the same objective functions, requiring only a change of the parameters $n$ and $\mu$.

In Fig. 2, a two-dimensional bifurcation diagram of $\lambda_1$ versus $\lambda_2$ is plotted using the successfully detected bifurcation parameter sets $\lambda = g_b(h_{1\text{end}}) = (g_{b1}, g_{b2})$. Successful detection means that the detected bifurcation parameters and the periodic point composed of the bifurcation parameters satisfied both $F_{\text{bif}}(g_b(h_{1\text{end}})) < C_{\text{bif}}$ and $F_{\text{pp}}(g_p(h_{2\text{end}})) < C_{\text{pp}}$. This figure confirms the accurate detection of both the period-doubling bifurcation parameters and the saddle-node bifurcation parameters for all numbers of periods by the NLPSO-based detection method. Note that NLPSO detected the bifurcation parameters both stable periodic points and unstable periodic points without any careful initialization.

Table IV shows the results of $300 (= R_{\text{max}})$ simulations initialized in different random states for each bifurcation type and periodic number. The table shows the means and standard deviations (SDs) of $F_{\text{bif}}(g_b(h_{1\text{end}}))$ and $F_{\text{pp}}(g_p(h_{2\text{end}}))$ and the mean $h_{1\text{end}}$ of the PSO$_{\text{bif}}$. In this table, Suc[%] values of the PSO$_{\text{bif}}$ and the PSO$_{\text{pp}}$ denote the successful detection rate. The PSO$_{\text{bif}}$ detected 100% of the period-doubling and saddle-node bifurcations, and the mean $h_{1\text{end}}$ was well below the iteration limit ($H_1 = 300$). Notably, the success rate of the PSO$_{\text{pp}}$ was also 100%. These results mean that NLPSO successfully detected the local bifurcations with 100%. Moreover, the small SDs of $F_{\text{bif}}$ and $F_{\text{pp}}$ confirm the high robustness of NLPSO. Although NLPSO does not guarantee that a global optimum can be found, there is a high probability an acceptable solution will be found. Therefore,
even if NLPSO failed to search for a solution, NLPSO can find the accurate bifurcation point with high probability by re-searching with the different random initial state. These facts show that NLPSO requires no careful initialization as the Newton-based method.

In considering the NLPSO optimization process, Fig. 3 shows the movement of the PSO bif particles that converged to the 2-periodic saddle-node bifurcation parameter $\lambda_1 - \lambda_2 = (0.4862, 0.5969)$. The yellow circles and red cross markers in the chart denote the position vectors $z_b(h_1)$ of 30 particles in the PSO bif and the $g_{best}$ $g_b(h_1)$ at the iteration count $h_1$, respectively. Since the particle position $z_b$ of the PSO bif corresponds to $\lambda$ in the circle map as described in Eq. (24), the particles move on a system parameter space of the $\lambda_1 - \lambda_2$ plane. Gray lines denote the 2-periodic saddle-node bifurcation curves [1]. In other words, the multiple optimum solutions, in which $F_{\text{bif}}(\lambda) = 0$, co-exist on the gray lines. On the other hand, Fig. 4 shows a search space of the PSO pp for the PSO bif $g_{best}$ at the iteration count $h_1$, and this figure is a return map of the circle map onto $x(k) - x(k + 2)$ plane for the system parameter $\lambda$ that is the PSO bif $g_{best}$ $g_b(h_1)$ at $h_1$. Black markers denote the 2-periodic orbit $x_p^* = g_p$ detected by the PSO pp. Since the particle position $z_p$ of the PSO pp corresponds to
Fig. 4. Return map of the circle map onto the $x(k)-x(k+2)$ plane for the parameter set $\lambda$ that is the \textit{gbest} $g_p(h_1)$ of the PSO$_{bif}$ at the iteration count $h_1$. Black circle denotes the 2-periodic point $x^*_p = g_p$ detected by the PSO$_{pp}$. 
(a) $(\lambda_1, \lambda_2) = (0.5313, 0.9854)$. $g_p = 0.11585$. (b) $(\lambda_1, \lambda_2) = (0.5263, 0.8447)$. $g_p = 0.18687$. (c) $(\lambda_1, \lambda_2) = (0.4831, 0.6656)$. $g_p = 0.40586$. (d) $(\lambda_1, \lambda_2) = (0.4862, 0.5969)$. $g_p = 0.40970$.

$x(0)$ in Eq. (24), the particles move on a line of the state variable $x$. In the initial state of the PSO$_{bif}$ at $h_1 = 0$ (Fig. 3(a)), the PSO$_{bif}$ particles were randomly spread in the target parameter space. In this way, we do not need to carefully set the initial value to search, but we only need to set it randomly. As the iterations proceeded, the particles swarmed toward one of the $G^2$ bifurcation parameters as shown in Figs. 3(b)–(c). Meanwhile, Figs. 4(a)–(c) show that a shape of optimization space of $F_{pp}$ varies with the $g_b(h_1)$ value. This shows that the PSO$_{pp}$ should be used for each PSO$_{bif}$ particle since the PSO$_{bif}$ particles take a different periodic point $x^*_p$ depending on respective particle position $z_{bi}$. Furthermore, the PSO$_{pp}$ found both SPPs and UPPs without careful initialization. If multiple SPPs and/or UPPs coexist in the map, the PSO$_{pp}$ particles converge to one of the periodic points, depending on the PSO$_{pp}$ initial state and the random number values in the updated equation (Eq. (2)). In the searching process, if a PSO$_{bif}$ particle takes parameters for which no 2-periodic point exists, the PSO$_{pp}$ returns a \textit{false} periodic point that lies outside the criterion $C_{pp}$ for that particle. Then, the PSO$_{bif}$ assigns a bad $F_{bif}$ value ($\infty$) to that particle. Since PSO steers the poor-scoring particles toward those with good $F_{bif}$ values, the PSO$_{bif}$ particles automatically move to the parameter area containing the correct 2-periodic points, without requiring special manipulation or careful initialization. Furthermore, since PSO$_{bif}$ particles located outside the target parameter space cannot become its \textit{pbest}, they are attracted back to the target parameter space in subsequent iterations. At $h_1 = 61$ (Fig. 3(d)), the PSO$_{bif}$ that found a saddle-node bifurcation point satisfied
the $F_{\text{bif}} < 10^{-3}$ criterion, which terminated the bifurcation point searching. Obviously, the PSO$_{\text{bif}}$ particles located outside the target parameter space never become the $\text{gbest}$. Therefore, the detected bifurcation parameter certainly locates inside the target parameter space.

### 5.2 Non-autonomous system — Duffing equation

Next, we apply NLPSO to the Duffing equation, which is a non-autonomous system and exhibits chaotic behavior. The system is given by

\[
\begin{align*}
\frac{dx_1}{dt} &= x_2 = f_1(t, x, \lambda), \\
\frac{dx_2}{dt} &= -0.2x_2 - x_1^3 + \lambda_2 + \lambda_1 \cos(t) = f_2(t, x, \lambda).
\end{align*}
\]

where $x = (x_1, x_2) \in \mathbb{R}^2$ is a vector of state variables, and $\lambda = (\lambda_1, \lambda_2) \in \mathbb{R}^2$ is a vector of system parameters. Equation (29) periodically varies according to the time $t$ with period $T = 2\pi$, so the Poincaré section is at $t = 2\pi$.

The objective function of the PSO$_{\text{bif}}$ is described by

$$F_{\text{bif}}(z_b) = \begin{cases} 
\det \left( \frac{\partial \Phi}{\partial \alpha} \left( x_p^* \right) - \mu I_N \right), & \text{if } F_{\text{pp}} \left( x_p^* \right) < C_{\text{pp}} \\
\infty, & \text{otherwise},
\end{cases}$$

where $z_b = (z_{b1}, z_{b2})$ is the particle position vector of the PSO$_{\text{bif}}$ and serves $\lambda$ in Eq. (29); thus, $L = 2$. The initial value $x_0(= x(0))$ is a deemed $n$-periodic point $x_p^* = (x_{p1}^*, x_{p2}^*)$ depending on the parameter set $(z_{b1}, z_{b2})$ and it is detected by the PSO$_{\text{pp}}$ with $z_b$. The system equation and the state variables $x$ can be described as

\[
\begin{align*}
\frac{dx}{dt} &= f \left( t, x, z_b \right), \\
x(t) &= \varphi \left( t, x_p^*, z_b \right).
\end{align*}
\]

Thus,

$$\varphi \left( 0, x_p^*, z_b \right) = x_p^*.$$

The elements of Jacobian matrix in Eq. (30) can be obtained by solving following ODEs with the system equations in Eq. (29):

\[
\begin{align*}
\frac{d}{dt} \left. \begin{bmatrix} \frac{\partial \varphi_1}{\partial x_{p1}^*} \\ \frac{\partial \varphi_2}{\partial x_{p1}^*} \end{bmatrix} \right|_{x = x_p^*} &= \left. \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{bmatrix} \right|_{x = x_p^*}, & \frac{d}{dt} \left. \begin{bmatrix} \frac{\partial \varphi_1}{\partial x_{p1}^*} \\ \frac{\partial \varphi_2}{\partial x_{p1}^*} \end{bmatrix} \right|_{x = x_p^*} &= \left. \begin{bmatrix} \frac{\partial f_1}{\partial x_{p1}^*} \\ \frac{\partial f_2}{\partial x_{p1}^*} \end{bmatrix} \right|_{x = x_p^*} - 3x_{p1}^* \varphi_2 - 0.2 \frac{\partial \varphi_2}{\partial x_{p1}^*} \\
\frac{d}{dt} \left. \begin{bmatrix} \frac{\partial \varphi_1}{\partial x_{p2}^*} \\ \frac{\partial \varphi_2}{\partial x_{p2}^*} \end{bmatrix} \right|_{x = x_p^*} &= \left. \begin{bmatrix} \frac{\partial f_1}{\partial x_{p2}^*} \\ \frac{\partial f_2}{\partial x_{p2}^*} \end{bmatrix} \right|_{x = x_p^*} - 3x_{p2}^* \varphi_1 - 0.2 \frac{\partial \varphi_1}{\partial x_{p2}^*} \end{align*}
\]

404
\[
\begin{bmatrix}
\frac{\partial \phi_1}{\partial x_1^{\ast}} \\
\frac{\partial \phi_1}{\partial x_1} \\
\frac{\partial \phi_2}{\partial x_1^{\ast}} \\
\frac{\partial \phi_2}{\partial x_1}
\end{bmatrix}_{t=0} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix},
\]

Similarly, the objective function of the \( \text{PSO}_{pp} \), which detects \((\lambda_1, \lambda_2)\)-dependent \( n \)-periodic solutions, is given by

\[
F_{pp}(z_p) = \| \Phi^n (z_p) - z_p \| = \sqrt{(\varphi_1 (2n\pi, z_p, \lambda) - z_{p1})^2 + (\varphi_2 (2n\pi, z_p, \lambda) - z_{p2})^2},
\]

where \( z_p = (z_{p1}, z_{p2}) \) is the \( \text{PSO}_{pp} \) particle position vector and serves \( x_0 \) in Eq. (29); thus, \( N = 2 \). The system parameter vector \( \lambda \) in Eq. (35) corresponds to a position vector \( z_b \) of a single \( \text{PSO}_{bif} \) particle that invoked the \( \text{PSO}_{pp} \).

The experimental parameters of \( \text{PSO}_{bif} \) and \( \text{PSO}_{pp} \) are same as in Table III. For different number of periodic solutions \( (n = 1 \text{ and } 2) \), the NLPSO detected a period-doubling bifurcation parameter \((\mu = -1)\) and a saddle-node bifurcation parameter \((\mu = 1)\). The search-space range of the \( \text{PSO}_{bif} \) is set as

\[
S_b : 0.4 \leq z_{b1} \leq 0.6 \land 0.5 \leq z_{b2} \leq 2.0, \text{ for } I^n
\]

\[
S_b : 0.2 \leq z_{b1} \leq 0.5 \land 0.5 \leq z_{b2} \leq 2.0, \text{ for } G^n
\]

The initialization range of \( z_p \) is set as \([-0.4, 0.4]^N\). We computed these bifurcation parameters in 300 simulations with different random initial states of the particles.

Figure 5 is a two-dimensional bifurcation diagram of the respectively detected 300 bifurcation parameter sets [11]. This figure confirms that the bifurcation parameter sets are obtained correctly on both the period-doubling and saddle-node bifurcation parameters for all numbers of periods because the detected parameters are located on the bifurcation curves detected by Kawakami method [1] as shown in Fig. 6. Table V summarizes the results of 300 simulations of each bifurcation type and periodic number. The success rate of both \( \text{PSO}_{bif} \) and \( \text{PSO}_{pp} \) were 100%, and the mean \( h_{1\text{end}} \) was below the iteration limit \((H_1 = 300)\). The small SDs of \( F_{bif} \) denotes the robustness of NLPSO. We

\[\text{Fig. 5. Parameter detection results of period-doubling bifurcations (\( I^n \)) and saddle-node bifurcations (\( G^n \)) in the Duffing equation. Three hundred points each. } \mu = -1 \text{ or } 1, n = 1 \text{ or } 2.\]
should mention that the obtained bifurcation points shown in Fig. 5 are ununiformly distributed. This is caused by a PSO characteristic that the particles are attracted to a search area where solutions that can be easier found exist. By limiting the search-space range of the PSO bif to the area we want to obtain and combining the bifurcation parameters detected with different search-space range, the NLPSO can obtain uniformly distributed bifurcation parameters.

6. Conclusions

This paper explained in detail the NLPSO-based detection method of the 2-parameter bifurcation diagram in a discrete-time dynamical system and a non-autonomous system. Since this method does not use the derivative of the objective function, it requires no careful initialization or Lyapunov exponents. It is particularly worth noting that the NLPSO-based method does not require second-order partial derivative in non-autonomous systems. It means the high versatile usability of the NLPSO. In contrast to the traditional Newton method-based approaches, our NLPSO-based method can detect bifurcation parameters of nondifferentiable/discontinuous systems since PSO can realize global search and does not require differentiability of objective functions. These features suggest that the NLPSO-based method will be developed into an efficient tool for detecting high-dimensional periodic points in high-dimensional systems with large parameter spaces. Furthermore, NLPSO can detect the bifurcation parameter sets at which stable and unstable periodic point merges. The NLPSO can make a 2-parameter bifurcation diagram by itself, however, by assigning the bifurcation parameter, detected by the NLPSO, as the initial point of a conventional curve tracing, we could solve the initial value problem of the conventional bifurcation analysis.

We hope to extend the NLPSO into the bifurcation analysis in various dynamical systems, circuit design, chaos control, and other applications. This study has used the standard PSO since this type
of PSO is the simplest and includes only a fraction of parameters. However, we should compare the standard PSO used in this study to modified PSOs including deterministic PSO [14–16] or other metaheuristic optimizers [17], such as the Nelder-Mead simplex method and differential evolution [18]. Moreover, we also would like to develop a NLPSO-based bifurcation curve tracing method and a MATLAB/Python toolbox or software to carry out our method. It would be a great help to researchers specializing in the study of the dynamical systems.

Acknowledgments

We also thank Dr. Y. Miino for lending his expertise on making the bifurcation diagram by the Kawakami method. This work was supported by JSPS KAKENHI Grant Numbers 19K12138, 19K12106 and 19K12144. This research was supported by a grant from the Hibi Science Foundation.

References

[1] H. Kawakami, “Bifurcation of periodic responses in forced dynamic nonlinear circuits: Computation of bifurcation values of the system parameters,” IEEE Trans. on Circuits and Systems, vol. 31, no. 3, pp. 248–260, 1984.
[2] T. Ueta, M. Tsueike, H. Kawakami, T. Yoshinaga, and Y. Katsura, “A computation of bifurcation parameter values for limit cycles,” IEICE Trans. on Fundamentals of Electronics, Communications and Computer Sciences, vol. E80-A, no. 9, pp. 1725–1728, 1997.
[3] T. Kousaka, T. Ueta, and H. Kawakami, “Bifurcation of switched nonlinear dynamical systems,” IEEE Trans. on Circuits and Systems II, vol. 2, no. 7, pp. 1451–1458, 1999.
[4] K. Tsumoto, T. Ueta, T. Yoshinaga, and H. Kawakami, “Bifurcation analyses of nonlinear dynamical systems: From theory to numerical computations,” NOLTA, vol. 3, no. 4, pp. 458–476, 2012.
[5] E.J. Doedel and B.E. Oldeman, 2012. http://indy.cs.concordia.ca/auto/.
[6] F. Dercole and Y.A. Kuznetsov, “SlideCont: An auto97 driver for bifurcation analysis of Filippov systems,” ACM Transactions on Mathematical Software (TOMS), vol. 31, no. 1, pp. 95–119, 2005.
[7] ERATO, JST, 2004. http://bunki.ait.tokushima-u.ac.jp:50080/.
[8] J. Kennedy and R. Eberhart, “Particle swarm optimization,” Proc. of IEEE Int. Joint Conf. on Neural Netw., pp. 1942–1948, 1995.
[9] H. Matsushita, Y. Tomimura, H. Kurokawa, and T. Kousaka, “Period Doubling Bifurcation Point Detection Strategy with Nested Layer Particle Swarm Optimization,” International Journal of Bifurcation and Chaos, vol. 27, no. 7, 1750101, 2017.
[10] H. Matsushita, H. Kurokawa, and T. Kousaka, “Saddle-node bifurcation parameter detection strategy with nested-layer particle swarm optimization,” Chaos, Solitons & Fractals, vol. 119, pp. 126–134, 2019.
[11] H. Matsushita, H. Kurokawa, and T. Kousaka, “Nested-layer particle swarm optimization method for bifurcation point detection in non-autonomous systems,” NOLTA, vol. 10, no. 3, pp. 289–302, 2019.
[12] Y. Shi and R. Eberhart, “A Modified Particle Swarm Optimizers,” Proc. of IEEE Int. Conf. on Evolutionary Computation, pp. 69–73, 1998.
[13] I.C. Trelea, “The particle swarm optimization algorithm: convergence analysis and parameter selection,” Information Processing Letters, vol. 85, no. 6, pp. 317–325, 2003.
[14] C.W. Cleghorn and A.P. Engelbrecht, “A generalized theoretical deterministic particle swarm model,” Swarm Intelligence, vol. 8, no. 1, pp. 35–59, 2014.
[15] Y. Yamanaka and T. Tsubone, “An optimizer using swarm of chaotic dynamical particles,” NOLTA, vol. 6, no. 1, pp. 112–130, 2015.
[16] T. Sasaki, H. Nakano, A. Miyachi, and A. Taguchi, “Deterministic particle swarm optimizer with the convergence and divergence dynamics,” IEICE Transactions on Fundamentals of Elec-
tronics, Communications and Computer Sciences, vol. 100, no. 5, pp. 1244–1247, 2017.

[17] I. Boussaïd, J. Lepagnot, and P. Siarry, “A survey on optimization metaheuristics,” Information Sciences, vol. 237, pp. 82–117, 2013.

[18] R. Storn and K. Price, “Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces,” Journal of Global Optimization, vol. 11, no. 4, pp. 341–359, 1997.