Article

Predominance Preferential Selection for Minimizing Surplus Parts in the Selective Assembly of a Flow Production System

Kanghyeon Shin , Hyobin Son , Kyohong Jin *

Posted Date: 31 December 2024

doi: 10.20944/preprints202412.2449.v1

Keywords: Keywords: Prioritization; Selective assembly; Surplus Parts; Flow production; Ball bearing

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 31 December 2024 doi:10.20944/preprints202412.2449.v1

Article

Predominance Preferential Selection for Minimizing Surplus Parts in the Selective Assembly of a Flow Production System

Kanghyeon Shin [1], Hyobin Son [2] and Kyohong Jin [3,*]

[1] DavinSoft Inc., Changwon 51397, Republic of Korea; shinkh@davinsoft.com

[2] Department of Smart Manufacturing Engineering, Changwon National University, Changwon 51140, Republic of Korea; wade99hb@gmail.com

[3] Department of Electronic Engineering, Changwon National University, Changwon 51140, Republic of Korea; khjin@changwon.ac.kr

* Correspondence: khjin@changwon.ac.kr; Tel.: +82-55-213-3656

Abstract:We propose the predominance preferential selection (PPS) algorithm as a successor to the density-based prioritization (DBP) approach to improve the quality and computational efficiency of selective assembly processes. The PPS algorithm achieves significant performance improvement by sorting components within slots according to their dimensional similarities and grouping them to assign search priorities. Additionally, it reduces the computational time using a narrowed search space and improves the management of surplus parts by maintaining dimensional diversity within the slots. In this study, we test the PPS algorithm using real process data obtained from a precision bearing manufacturer and demonstrate its efficiency in improving matching rates while maintaining statistical reliability. Additionally, by making minimal modifications, we demonstrate its adaptability to multiple-slot row structures in a wide range of applications. Although the process capability indices exhibited a slight decrease, the overall efficiency and matching rate improvement were evident.

Keywords: Prioritization; Selective assembly; Surplus Parts; Flow production; Ball bearing

1. Introduction

Selective assembly is a manufacturing process for improving the precision of assembled components, particularly in environments where part variations can significantly affect product quality and functionality. Selective assembly is a process used to carefully match parts based on their dimensional characteristics to compensate for machining errors, and products such as ball bearings and pistons are representative products that typically utilize selective assembly [1]; These products are essential across various industries, from small machinery to the automotive and aerospace sectors. For components used in the automotive and aerospace industries, which require high precision, selective assembly plays a crucial role in achieving these stringent requirements. Despite its advantages, selective assembly exhibits some drawbacks, notably in managing surplus parts and ensuring high process capability within the confinement of assembly tolerances [2].

Algorithms and strategies have been developed from various perspectives to optimize the selective assembly process, and the recent research trends and related research are as follows. For example, the selective assembly process can be treated as an optimization problem. A representative approach involves using meta-heuristic, represent as genetic algorithms (GAs), which rely on biologically inspired operators, such as selection, crossover, and mutation, to provide high-quality solutions to optimization and search problems. Kannan and Raja Pandian employed a GA in the selective assembly of engine components (pistons, piston rings, and cylinders) [3]; they combined selective groups based on the dimensional distribution of the quality characteristics of the above three components. Subsequently, they constructed the GA into five modules—Input, Evaluation, Initialization, New Population Generation, and Output—to determine the optimal combinations of components. Xing and Wang proposed a flowchart for optimizing selective groups based on an advanced GA referred to as IGAOT [4]; they coded the group structure into six pairs of chromosomes and applied a GA that ranks fitness values based on simulated annealing to optimize chromosome group combinations using the objective function as a basis.

In addition, I would like to introduce the papers of Lu, J., Yang, L., Huang, Y., Long, J., Li, C., and Yang, Z as the recent paper on selective assembly optimization through GA, proposed an optimization method (CCOM) for maintaining clearance consistency in small batch bearing assembly [5]; The study focuses on minimizing the loss of components caused by manufacturing errors while simultaneously achieving high assembly success rates and clearance consistency. Utilizing a genetic algorithm, the method identifies optimal combinations, and its validity was experimentally verified using NCF3044 bearing data. This research contributes to reducing surplus component issues and improving assembly efficiency in selective assembly processes.

Xiao et al. proposed an assembly selection and matching methodology aimed at improving the performance of remanufactured mechanical products, constructing a multi-objective optimization model utilizing the Taguchi quality loss function and remanufacturing cost function.[6] In particular, a method combining Particle Swarm Optimization (PSO) and Genetic Algorithm (GA) was employed to comprehensively optimize resource utilization efficiency, assembly precision, and production cost during the assembly process. Using the assembly process of the CAK6140 machine tool spindle box as a key case study, the proposed methodology demonstrated its effectiveness when compared to existing methods. This study addressed uncertainties and cost issues in traditional remanufacturing assembly processes and enabled the improvement of product quality and performance in remanufactured products.

The above studies adopted machine learning model performance metrics, such as the coefficient of determination (R^2) and root mean square error, for algorithm evaluation; however, these metrics do not take into account the reduction in processing time [7] because of the inherent nature of GA-based algorithms. These algorithms require multiple generations in a single computation, which restricts their ability to maximize computational efficiency.

There have been a number of studies in terms of statistical distribution and cost reduction of parts dimensions, not meta-heuristic. Recently, Andrea Mencaron et al. proposed a hybrid assembly method to address the high operational costs associated with selective assembly [8]; This study focused on designing cost-effective and optimal matching strategies by utilizing the statistical distribution of component dimensions instead of relying on meta-heuristic approaches. The proposed method reduces assembly costs by transitioning unmatched components from the traditional assembly process to selective assembly. This study introduced a cost function to quantitatively analyze the balance between traditional and selective assembly costs while statistically evaluating the impact of changing matching probabilities on cost reduction. In a case study involving shaft and hole assembly, the hybrid approach demonstrated a cost reduction of approximately 45.8% compared to conventional selective assembly methods. However, this method has a limitation that it is difficult to apply to processes with low flexibility in process structure.

Zhang, Yin, and Yang designed a selective assembly algorithm that deals with the design tolerances of holes and shafts [9]; they categorized the dimensional boundaries of each shaft and hole into groups of nonbasic mating sizes and further subdivided them into clearance fit, interference fit, and transition fit. Depending on the case, they applied specific process techniques such as remachining, measurement, assessment, and marking).

Filipovich proposed a simulation model for selective assembly [10] and introduced a block diagram of the information-control system to optimize the process and improve accuracy [11]; by employing a selective assembly algorithm and a complementary five-step block diagram comprising a measuring device, an estimation subsystem, a control subsystem, a communication channel, an output device, and an assembly machine, error reduction and efficiency improvement were achieved.

Fang and Zhang presented algorithms based on statistical figures and the z-group to increase matching accuracy [12]. The above studies focused on single-parameter selective assembly, indicating the need for expanding this framework to multiparameter assembly systems.

Shin and Jin proposed the density-based prioritization (DBP) algorithm to minimize surplus parts by prioritizing the selection of parts according to their dimensional density [13]; This algorithm has significant benefits in increasing the matching rate while reducing the computational complexity of the assembly process. This is suitable for a process where parts are continuously fed, such as ball bearing, and a combination of parts is selected in a short time to produce the assembly result. however, a critical limitation of this algorithm is its lack of considering how close the assembled dimensions are to the ideal dimensions. This can lead to a uniform distribution of assembled dimensions within the tolerance band, significantly limiting the process capability, which is a measure of a process' ability to produce parts within specified tolerances.

Considering the inherent limitations of the DBP approach, in this study, we introduce the predominance preferential selection (PPS) algorithm; this algorithm addresses the surplus parts issue and improves process capability. Unlike the DBP algorithm, the PPS algorithm incorporates a novel technique of grouping priorities into specific units, treating them as equivalent in priority. This methodology ensures that from the available combinations of parts with the same priority, the combination of parts with dimensions closest to the ideal dimensions is selected. Such an approach naturally resolves the above-mentioned process capability problem by promoting a distribution of assembled dimensions that are tighter and more aligned with the ideal dimensions.

The PPS algorithm can also be applied to multiple-slot row structures because of its improved versatility and scalability compared with other algorithms. In contrast, the DBP algorithm, which was designed for assembly facilities employing one slot row, exhibits difficulties in prioritizing parts across multiple components because of the conflicting priority orders in assembly facilities employing more than one slot row. By grouping priorities, the PPS algorithm seamlessly integrates multiple priority orders into a unified sequence; thus, it can be applied to complex assembly facilities that feature multiple-slot rows. Its scalability is particularly important in modern manufacturing lines, which increasingly rely on the efficient assembly of multiple components to meet high precision and quality standards.

To empirically validate the efficiency of the proposed PPS algorithm, we simulated a selective assembly scenario that mirrors real-world conditions. The simulation was based on data collected from an actual bearing assembly facility, where three critical components are assembled. The facility layout, which features slot rows for two out of the three components, provides a complex but highly relevant environment for applying the PPS algorithm. Our objective is to meticulously assess the impact of the PPS algorithm on two key performance indicators: surplus part ratio and process capability.

2. Selective Assembly Employing Multiple-Slot Rows in Flow Production Systems

Selective assembly is particularly important in the assembly of high-precision products, such as bearings, where minimal deviations can significantly affect product performance. In this section, we present the selective assembly process in a bearing assembly facility equipped with multiple-slot rows, focusing on specific challenges encountered therein.

2.1. Selective Assembly Process

In the investigated bearing assembly process, the component specifications are defined as follows (Figure 1):

- Outer ring raceway diameter: $A = 40.017 \pm 0.020$ mm;
- Inner ring raceway diameter: $B = 24 \pm 0.010$ mm;
- Ball diameter: $C = 8 \pm 0.0005$ mm;

doi:10.20944/preprints202412.2449.v1

- Assembly clearance: Y = 0.017 ± 0.005 mm.

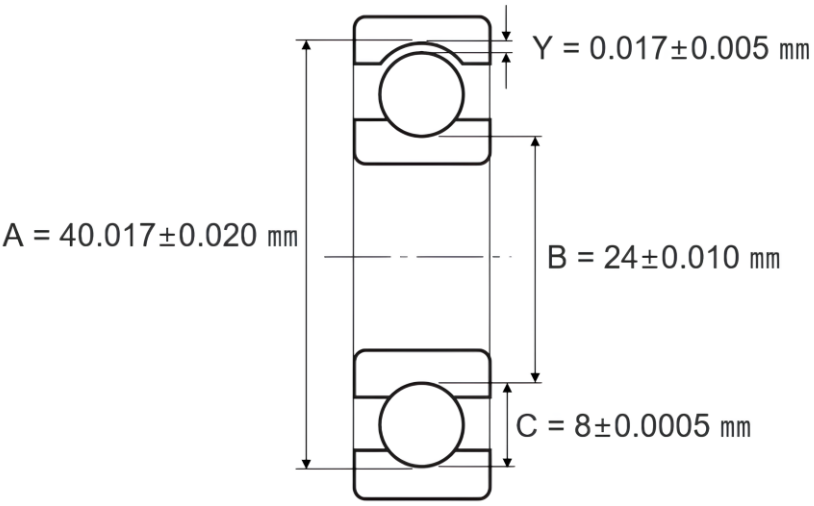


Figure 1. Precision ball-bearing assembly.

To achieve precision in the assembly process, it is important to calculate the assembly clearance Y. Tolerance (clearance) Y is the target parameter, which is commonly adopted in general selective assembly algorithms, to evaluate component compatibility [14]. In this case, Y is determined using the following equation:

$$Y = A-B-2C \tag{1}$$

The above equation indicates the interdependence of the component dimensions to achieve the target assembly clearance. Given the specified tolerances for A, B, and C, the above equation assesses component compatibility and ensures that the assembled bearing meets the required precision standards. Y represents the cumulative effect of variations in the component dimensions within its corresponding tolerances.

Given these tolerances, for simplification, the basic dimensions are offset and omitted during the selective assembly process. Therefore, the tolerances are redefined without specifying the nominal dimensions; this is a common practice when focusing on component variability and compatibility rather than its absolute dimensions:

- Outer ring raceway diameter tolerance: A = ±20 μm;
- Inner ring raceway diameter tolerance: B = ±10 μm;
- Ball diameter tolerance: C = ±0.5 μm;
- Assembly clearance tolerance: Y = ±5 μm.

The assembly facility employs two slot rows including 10 slots for the outer rings and 6 slots for the inner rings along with 3 ball tanks, which are categorized according to the diameter biases of the balls. The diameter biases of the balls allocated to each tank are as follows:

$$C_{-} = -3 \pm 0.5\ \mu\text{m}$$

$$C_{0} = 0 \pm 0.5\ \mu\text{m}$$

$$C_{+} = 3 \pm 0.5\ \mu\text{m}$$

The three sizes of the balls in the tanks—each with a different diameter bias (C_{-}, C_{0}, and C_{+})—are strategically selected during the selective assembly process to correct the assembly clearance, thus increasing the probability of a successful assembly. This setup enables the system to allow for component dimensional variations using balls with slight size variations, thus improving the compatibility among outer rings, inner rings, and balls.

The bearing assembly process, which is depicted in Figure 2, follows a structured flow production system. In each production cycle, one outer ring and one inner ring are supplied to the assembly facility. The balls, which are preproduced with specific diameter biases, are stored in three separate tanks. During assembly, potential combinations based on the dimensions of the outer rings, inner rings, and balls stored in the tanks are calculated. Using Equation (1), the assembly system evaluates each combination of rings and balls to identify those that satisfy the Y = ±5 μm clearance.

If multiple combinations satisfy this tolerance, the system selects the combination with the smallest deviation from the ideal 0.017 mm clearance. This selection criterion ensures that the resulting tolerance is as close as possible to the ideal tolerance, thus optimizing the precision of the assembled product.

Upon selecting the best combination, the assembly facility completes the assembly process, and the now-vacant slots are refilled with the newly supplied outer and inner rings. This continuous refilling process ensures that the outer and inner ring slots remain fully occupied, allowing the assembly process to proceed without interruption according to the production system flow.

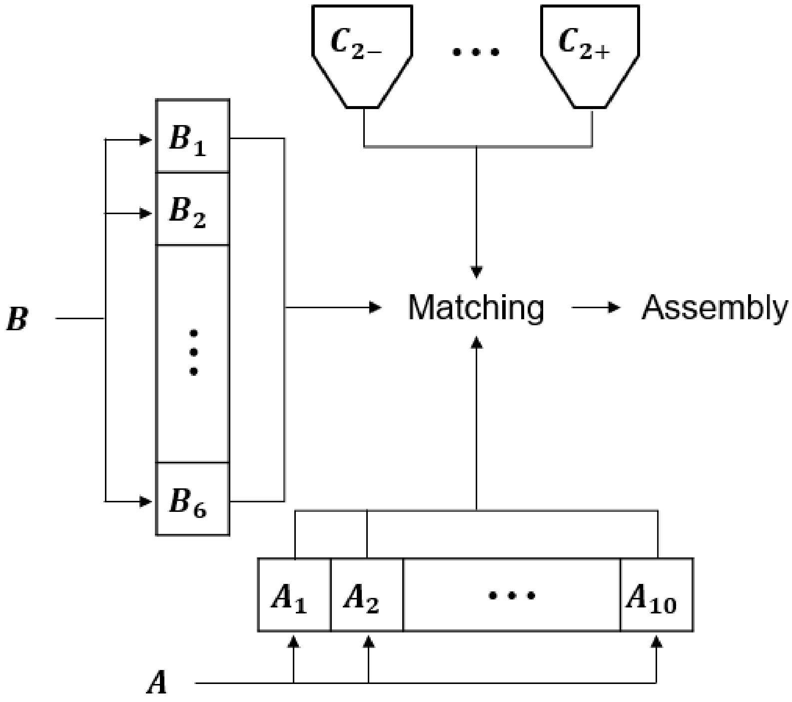


Figure 2. Selective assembly of the precision ball bearing.

If no combination meets the assembly tolerance, all outer rings in the slots are emptied and replenished with the newly supplied outer rings. The removed outer rings are either reprocessed or discarded as surplus parts. Due to the higher precision in the machining of the inner rings than that in the outer rings, the inner rings are not discarded; instead, they remain in the slots for use in subsequent assembly cycles.

The generation of surplus parts has a dual impact: 1) leads to material waste and 2) temporarily halts the assembly process until all slots are refilled. This pause reduces production efficiency and increases the overall manufacturing cost because it disrupts the continuous flow required for high productivity in flow production systems.

2.2. Cause of Surplus Parts

As the process progresses, the number of surplus parts that are structurally accumulated in the slots increases. This is due to two main factors: 1) The deviation in the dimensional distribution of the raceway diameter caused by machine wear during the grinding process. Although the deviation

doi:10.20944/preprints202412.2449.v1

in the dimensions of the parts fed into the assembly machine remains below the threshold, the dimensions of the produced components deviate from the design dimensions. This deviation causes an uneven dimensional distribution in the components, leading to a deviation in the process dimensions of each component.

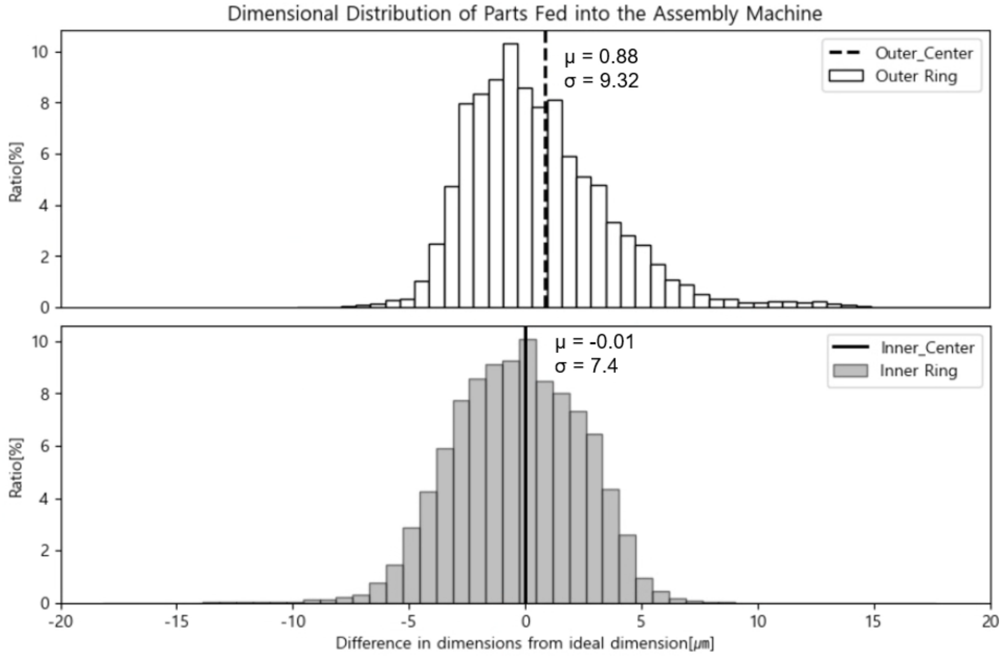
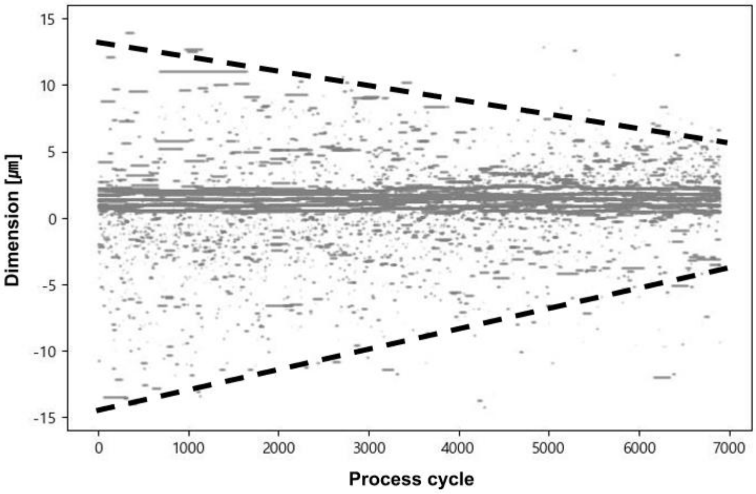
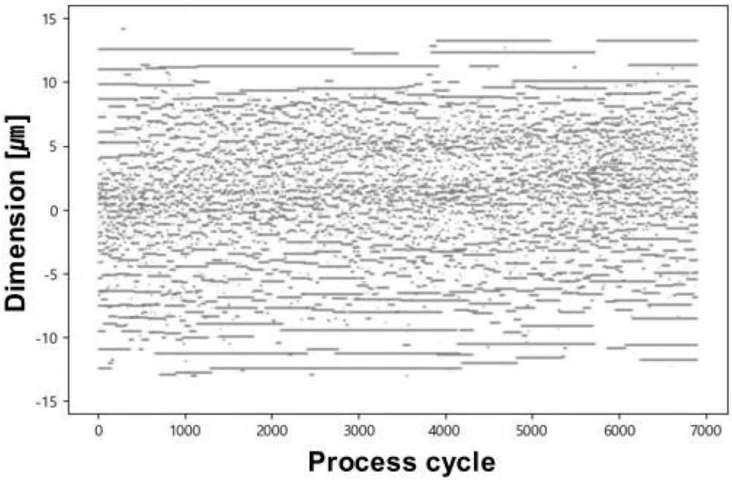


Figure 3. Dimensional Distribution of parts fed into the assembly machine.

2) The matching algorithm method employed for component selection. In conventional approaches, the algorithm selects the combination of components that results in the assembly tolerance Y that is closest to the design tolerance value. Although this method ensures the best quality by considering all possible combinations, it potentially increases the number of surplus parts in the slots. Figure 4 shows the dimensional distributions of the outer and inner rings introduced into the assembler over the same length of a production cycle. Both distributions exhibit a near-normal distribution centered around their respective mean values. In an optimal scenario, components with dimensions near the center of their distribution are matched with each other, whereas components with large deviations are paired together.



(**a**)



(**b**)

Figure 4. a. Dimensional distribution in slots vs. process cycle in the ordinary case. b. Dimensional distribution in slots vs. process cycle in the optimal case.

However, in an ordinary process algorithm, which evaluates all combinations to derive the optimal result, even when alternative combinations of similar quality are available (e.g., components near the center of the distribution), the algorithm often disrupts the dimensional diversity in the outer and inner ring slots to achieve optimal quality. Figure 4-a shows the actual process data. We observe that as the production cycle progresses, the dimensional diversity within the slots converges near the center of the distribution. In contrast, the optimal production cycle is shown in Figure 4-b, where the dimensional diversity within the slots is maintained, ensuring a balanced and efficient assembly process over time.

3. PPS Algorithm

The proposed PPS algorithm was developed to improve the process capability and flexibility of selective assembly systems, particularly those with multiple-slot rows. In systems with multiple-slot rows—one for each of the multiple key components—PPS assigns and evaluates priorities across all rows to optimize component matching. Although PPS shares fundamental principles with DBP [13], it introduces distinctive features to improve compatibility and precision in the assembly process.

In PPS, priority rankings are assigned to the components stored in each slot row. For each row, the components are categorized according to their dimensional similarities; those that are closely clustered according to their dimensions are assigned higher priority in the assembly process than those not closely clustered. The dimensional similarities are calculated by first sorting the components in each row according to their size. Then, for each component, the size difference between its nearest smaller and bigger neighboring components is calculated. This difference represents the component dimensional similarity score; a small difference indicates a high similarity to other components. For components that fall in the middle of the sorted list, the dimensional similarity is the difference of the nearest neighboring component on both sides. However, for the smallest and largest components in the sorted order—those with only one neighboring component—the similarity score is calculated as twice the single size difference to approximate the effect of neighboring components on both sides.

Function: SortBySimilarity

```
function SortBySimilarity (array X)
    $n \leftarrow \text{length}(X)$
    initialize $S[1 \ldots n]$
    sort X in ascending
    for $i \leftarrow 2 \ldots n-1$ do
        $S[i] \leftarrow X[i+1]-X[i-1]$
    end for
    $S[1] \leftarrow 2 \times (X[2]-X[1])$
    $S[n] \leftarrow 2 \times (X[n]-X[n-1])$
    sort X by S in ascending
    return X
end function
```

An important concept in PPS is the “stride,” which allows priority rankings to be grouped into intervals, thereby enabling components within certain ranges to be treated as having equivalent priorities. This parameter, which can be set independently for each slot row, defines the size of these intervals and controls the number of components that share the same priority level.

For example, let us consider a row containing six parts categorized according to their dimensional priority, which is conventionally assigned the priority order in the 1–to 6 range. When applying a stride of 2, the six parts are grouped into three priority levels: two parts share priority 1, the next two share priority 2, and the last two share priority 3. If the stride is increased to 3, the six parts will be grouped into two priority levels with three parts assigned to each level {1, 1, 1, 2, 2, 2}. This configuration allows a flexible adjustment of “narrow” or “wide” priorities assigned within each row.

This stride-based grouping can be configured separately for each slot row. Consequently, each possible combination of components is assigned a combined priority, which is calculated as the sum of the priority levels assigned to individual components. By introducing stride, PPS establishes a structured priority hierarchy among component combinations, where several combinations share the same summed priority. Each set of combinations with identical priority levels is referred to as a ”block.”

PPS evaluates the assembly combinations by iterating through these priority blocks, starting with the highest priority block. Within each block, PPS searches for combinations that satisfy the assembly tolerance. If a suitable combination is found, it is selected for assembly. However, if multiple combinations within the same block meet the assembly tolerance, the algorithm refines its selection further by selecting the combination closest to the ideal dimension, thus minimizing deviation from the ideal dimension. This block-based selection strategy reduces the possibility of a uniform distribution of the assembled component clearances within the tolerance band. By prioritizing combinations that result in assembled component dimensions close to the optimal, PPS leads to a distribution that is more centered around the ideal clearance than that of other algorithms, thereby reducing variance.

Algorithm: Predominance Preferential Selection

```
A ← array of measurements of parts A
B ← array of measurements of parts B
strideA ← stride for array A
strideB ← stride for array B
A ← SortBySimilarity(A)
B ← SortBySimilarity(B)
n ← length(A)
m ← length(B)
initialize empty list P
for i ← 1…n do
 for j ← 1…m do
 p ← quotient(i / strideA) + quotient(j / strideB)
 if p ≥ length(P), then
 add empty list to P
 end if
 add tuple (i, j) to P[p]
 end for
end for
initialize empty list Q
for block in P do
 for i, j in block do
   r ← dimension when A[i] and B[j] are assembled
   if r satisfies tolerance, then
     e ← absolute of error between r and ideal dimension
     add tuple (e, i, j) to Q
   end if
 end for
 if Q is not empty then
   break
 end if
end for
select A[i], B[j] by i, j, where minimum e is from Q
```

The stride parameter is important in determining the granularity of priority rankings within each slot row. When stride is set to 1 for all slot rows, each component retains an individual priority ranking, and no blocks are formed. In this case, PPS operates similar to DBP, where the combinations are evaluated and selected strictly according to their priority order. Conversely, if stride equals the total slot capacity of a row, all components in that row will be grouped into a single priority level, effectively bypassing the need for priority distinctions and focusing purely on minimizing the deviation from the target dimension across all combinations. This flexibility allows PPS to be adjusted to varying assembly conditions and requirements by accommodating both narrow and wide prioritization ranges.

4. Performance Evaluation and Results

4.1. Testing Environment Evaluation Criteria

The PPS algorithm was tested using two-week real process data provided by a precision ball-bearing manufacturer. The raw data contained metadata including the timestamp of the received signal, the dimensions of the inner and outer rings, and the tolerance measurements. Additionally, the data contained information about the dimensions of the components within the slots at the time of observation. To validate the PPS algorithm, we compared the reconstituted ordinary process algorithm with two cases of the PPS algorithm using stride.

The operating principles of the ordinary process and PPS algorithms were omitted because they have been reported in previous studies. The results showed that the outerinner slot was 106 and the ball was classified into three types of barrels. The evaluation criteria included computation time, matching rate, C_{pk}, and adjusted matching rate; these are described below:

1. Computation time: it should be minimized to avoid time delay from combination selection to assembly.
2. Matching rate: whenever the process cycle is in progress, either the matching or the unmatching count increases, depending on whether the right group has been found. The matching rate is the percentage of the matching and unmatching ratios.
3. C_{pk}: C_{pk} (adjusted process capability index) is used to evaluate the consistency and quality of the assembled product according to the following equations:

$$\boldsymbol{C_p} = \frac{\boldsymbol{USL - LSL}}{\boldsymbol{6\sigma}} \tag{2}$$

$$\boldsymbol{C_{pk}} = \min\left(\frac{\boldsymbol{USL - \mu}}{\boldsymbol{3\sigma}}, \frac{\boldsymbol{\mu - LSL}}{\boldsymbol{3\sigma}}\right) \tag{3}$$

- USL: upper specification limit
- LSL: lower specification limit
- σ: standard deviation
- μ: clearance mean

4. Adjusted matching rate: This metric indicates the probability of defective components identified at the end of the process clearance stage, accounting for the inherent mechanical limitations. Due to limitations in the sensor accuracy, some dimensional errors in individual components may not be detected, which cannot be directly observed. To estimate this, potential errors (e.g., sensor errors, ball diameter errors, etc.) are randomly generated following a normal distribution and added to the process clearance variance. This simulates the latent errors during the process. Then, the adjusted matching rate, which represents the probability of a potential error, is calculated for a low C_{pk} considering the number of components that fall outside the clearance specification limits after incorporating these potential errors.

4.2. Simulation Results

The simulation results, which are presented in Table 1, confirm the efficiency of the PPS algorithm. The simulation, which was conducted by progressively reducing the stride from its largest value of 10–6 (same mechanism as that of the ordinary algorithm) to 2-2, showed a gradual increase in the matching rate. Although the decrease in C_{pk} may appear significant, the decrease in the adjusted matching rate, which is a statistical representation of C_{pk}, was not significant compared with the increase in the matching rate. Additionally, the computation time for the 2-2 stride decreased to 1/5 of the initial value, demonstrating that PPS effectively improves the matching rate, computation time, and scalability in multiple-slot row structures.

Table 1. Simulation Results.

	Ordinary algorithm	PPS (Stride 5-3)	PPS (Stride 2-2)
Computation time [ms]	0.060 ± 0.237	0.022 ± 0.150	0.010 ± 0.099
Matching rate [%]	95.11	96.54	97.09
C_{pk}	1.76	1.43	1.15
Adjusted matching rate [%]	95.10	96.50	97.00

5. Conclusions

As a follow-up to the DBP algorithm, we developed the PPS algorithm to improve the quality of selective assembly processes and optimize computation efficiency. PPS is based on sorting components within slots according to their dimensional similarities and grouping them to assign search priorities. PPS achieves a significant performance improvement by narrowing the search space to improve computational efficiency and ensuring dimensional diversity within slots to facilitate better management of surplus parts. Additionally, we introduced metrics to validate the potential drawbacks of the PPS algorithm, such as increased tolerance variance, to cross-verify its performance.

Apart from performance improvement, the main advantage of the PPS algorithm is its flexibility. Since PPS is based on component dimensional similarities for sorting and grouping, it can be widely applied to selective assembly processes employing slot structures and quality control metrics that are value- or dimension-based. Although this study focuses on a two-dimensional selective assembly process, the proposed PPS algorithm can be easily adapted to handle slot structures with more than two dimensions by making minimal modifications. In summary, PPS is a robust algorithm that achieves both computational efficiency and improved matching rates across a variety of selective assembly applications. Furthermore, its adaptability makes it suitable for addressing additional variables such as the integration of new assembly lines.

However, in this study, the PPS algorithm was not tested in real-time operations in an actual production environment. Although the algorithm incorporates priority-based search to reduce computation time, its feasibility for integration with factory PLCs was not validated. Furthermore, it was not tested in a wide range of manufacturing environments; therefore, additional testing and further research are required to address these limitations.

Author Contributions: Conceptualization: K.S, H.S., and K.J.; methodology: K.S, H.S., and K.J.: software: K.S. and H.S.; validation: K.S, H.S., and K.J.; formal analysis: K.S, H.S., and K.J.; investigation: K.S, H.S., and K.J.; resources: K.J. and H.S.; data curation: K.S. and H.S.; writing—original draft preparation: K.S, H.S., and K.J.; writing—review and editing: K.J. and H.S.; visualization: K.S. and H.S.; supervision: K.J.; project administration: K.J.; funding acquisition: K.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by "Regional Innovation Strategy" through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (MOE) (2021RIS-003). It was also supported by the Ministry of Science and ICT (MSIT), Korea, under the Information Technology Research Center support program (IITP-2024-RS-2024-00438409) supervised by the Institute for Information & Communications Technology Planning & Evaluation (IITP).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data were obtained from NSK Korea Co., Ltd. (Changwon Plant) and are available from the authors with the permission of NSK Korea Co., Ltd. (Changwon Plant).

Acknowledgments: This work was supported by the "Regional Innovation Strategy" program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education (MOE) (2021RIS-003). We also acknowledge the support from the Ministry of Science and ICT (MSIT), Korea, under the Information Technology Research Center support program (IITP-2024-RS-2024-00438409), supervised by the Institute for Information & Communications Technology Planning & Evaluation (IITP).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hui Liao, Pengfei Xie, Sier Deng, Wenhu Zhang, Lei Shi, Sen Zhao, Hengdi Wang,Research on early fault intelligent diagnosis for oil-impregnated cage in space ball bearing, Expert Systems with Applications, Volume 238, Part D,2024,121952,ISSN 0957-4174,https://doi.org/10.1016/j.eswa.2023.121952.

2. Wang, W.M.; Li, D.B.; He, F.; Tong, Y.F. Modelling and optimization for a selective assembly process of parts with non-normal distribution. *Int. j. simul. model.* **2018**, *17*, 133–146, doi:10.2507/ijsimm17(1)co1.
3. Kannan, S.M.; Raja Pandian, G. A new selective assembly model for achieving specified clearance in radial assembly. *Mater. Today.* **2021**, *46*, 7411–7417, doi:10.1016/j.matpr.2020.12.1229.
4. Xing, Y.; Wang, Y. Minimizing assembly variation in selective assembly for auto-body parts based on IGAOT. *Int. J. Intell. Comput. Cybern.* **2018**, *11*, 254–268, doi:10.1108/ijicc-10-2016-0039.
5. J. Lu, L. Yang, Y. Huang, J. Long, C. Li and Z. Yang, "A Clearance Consistency-Oriented Method for Small Batch Bearing Assembly," 2024 IEEE International Conference on Sensing, Diagnostics, Prognostics, and Control (SDPC), Shijiazhuang, China, 2024, pp. 153-157, doi: 10.1109/SDPC62810.2024.10707755.
6. Xiao, Y., Wang, Z., & Hu, S. (2024). A novel approach for assembly selection and matching of remanufactured mechanical products based on performance upgrading. Research Article. DOI: 10.21203/rs.3.rs-3767804/v1
7. Golinitskiy, P.; Antonova, U. Assessment of the influence of measurement error on the quality of selective assembly. *J. Phys.: Conf. Ser.* **2020**, *1515*, 052021, doi:10.1088/1742-6596/1515/5/052021.
8. Andrea Mencaroni, Dieter Claeys, Stijn De Vuyst, A novel hybrid assembly method to reduce operational costs of selective assembly, International Journal of Production Economics, Volume 264, 2023, 108966, ISSN 0925-5273,https://doi.org/10.1016/j.ijpe.2023.108966.
9. Zhang, Y.; Yin, Y.; Yang, M. A new selective assembly approach for remanufacturing of mating parts. In Proceedings of the The 40th International Conference on Computers & Indutrial Engineering; IEEE: Awaji, Japan, July 2010; pp. 1–6.
10. Filipovich, O. Simulation model of selective assembly of three elements. In Proceedings of the 2020 International Multi-Conference on Industrial Engineering and Modern Technologies (FarEastCon); IEEE: Vladivostok, Russia, October 6 2020; pp. 1–4.
11. Filipovich, O.; Maistrishin, M.; Filipovich, V. Model of information-control system used for one-parameter selective assembly of three elements. In Proceedings of the 2022 International Russian Automation Conference (RusAutoCon); IEEE: Sochi, Russian Federation, September 4 2022; pp. 876–880.
12. Fang, X.D.; Zhang, Y. A new algorithm for minimising the surplus parts in selective assembly. *Comput. Ind. Eng.* **1995**, *28*, 341–350, doi:10.1016/0360-8352(94)00183-n.
13. Shin, K.; Jin, K. Density-based prioritization algorithm for minimizing surplus parts in selective assembly. *Appl. Sci.* **2023**, *13*, 6648, doi:10.3390/app13116648.
14. Mahalingam, S.K.; Nagarajan, L.; Velu, C.; Kumar Dharmaraj, V.; Salunkhe, S.; Hussein, H.M.A. An Evolutionary Algorithmic Approach for Improving the Success Rate of Selective Assembly Through a Novel EAUB Method. *Appl. Sci.* **2022**, *12*, 8797, doi:10.3390/app12178797.