Table 3: Comparison of Relative Deviation from TCL, Improvement over GA, and CPU time taken by algorithms for Bomberger's problem

	% Relative Deviation from TCL		% Improvement over GA	CPU time (sec.)
Utilization (%)	**GA**	**Hybrid PSO**	**Hybrid PSO**	**Hybrid PSO**
50	1.308	1.276	0.031	15.189
55	1.776	1.766	0.009	15.019
60	2.505	2.456	0.048	15.489
65	3.403	3.405	0	15.690
66.18	4.234	4.234	0	15.972
70	7.160	7.160	0	16.059
75	9.656	9.721	0	16.060
80	10.979	10.834	0.130	15.561
83	11.414	11.414	0	16.205
86	13.868	12.945	0.811	14.813
88.24	15.727	15.725	0.001	13.786
89	16.544	16.547	0	13.465
92	26.327	30.743	0	11.131
95	42.751	41.939	0.569	11.075
97	51.829	51.752	0.051	11.283
98	56.450	55.964	0.311	11.140
99	85.503	58.806	14.392	11.063
Average	21.261	19.805	0.962	14.059
Min.	1.308	1.276	0	11.063
Max.	85.503	58.806	14.392	16.205
σ	23.939	20.042	3.469	2.078

Table 4: Detail comparison of GA and Hybrid PSO results for Bomberger's problem

Utilization	**Meta-heuristic**	
	GA	**Hybrid PSO**
50	T= 28.183 and K=[5,1,2,1,2,4,10,1,3,1]	T = 28.594 and K=[3,2,2,1,2,4,8,1,3,1]
55	T = 28.762 and K=[5,2,2,1,2,4,8,1,2,1]	T = 29.439 and K=[5,2,2,1,2,4,9,1,2,1]
60	T = 28.863 and K=[4,1,1,1,2,4,9,1,2,2]	T = 29.306 and K=[5,1,1,1,2,4,8,1,2,2]
65	T = 30.828 and K=[2,1,1,1,2,3,7,1,2,1]	T = 30.838 and K=[2,1,1,1,2,3,7,1,2,1]
66.18	T = 30.443 and K=[2,1,1,1,2,2,6,1,2,1]	T = 30.449 and K=[2,1,1,1,2,2,6,1,2,1]
70	T = 33.42 and K=[2,1,1,1,1,2,3,1,2,1]	T = 33.42 and K=[2,1,1,1,1,2,5,1,2,1]
75	T = 31.794 and K=[3,1,1,1,2,3,7,1,1,1]	T = 32.11 and K=[3,1,1,1,2,4,6,1,1,1]
80	T = 34.438 and K=[2,1,1,1,1,3,6,1,1,1]	T = 35.28 and K=[3,1,1,1,1,3,6,1,1,1]
83	T = 34.951 and K=[1,1,1,1,1,2,5,1,1,1]	T = 34.961 and K=[2,1,1,1,1,2,5,1,1,1]
86	T = 37.131 and K=[1,1,1,1,1,1,5,1,1,1]	T = 38.371 and K=[1,1,1,1,1,2,4,1,1,1]
88.24	T = 38.442 and K=[1,1,1,1,1,1,3,1,1,1]	T = 38.436 and K=[1,1,1,1,1,1,3,1,1,1]
89	T = 41.748 and K=[1,1,1,1,1,1,3,1,1,1]	T = 41.758 and K=[1,1,1,1,1,1,3,1,1,1]
92	T = 53.904 and K=[1,1,1,1,1,1,2,1,1,1]	T = 53.914 and K=[1,1,1,1,1,1,1,1,1,1]
95	T = 75.809 and K=[1,1,1,1,1,1,1,1,1,1]	T = 75 and K=[1,1,1,1,1,1,1,1,1,1]
97	T = 125.08 and K=[1,1,1,1,1,1,1,1,1,1]	T = 125 and K=[1,1,1,1,1,1,1,1,1,1]
98	T = 188.14 and K=[1,1,1,1,1,1,1,1,1,1]	T = 187.5 and K=[1,1,1,1,1,1,1,1,1,1]
99	T = 439.45 and K=[1,1,1,1,1,1,1,1,1,1]	T = 375 and K=[1,1,1,1,1,1,1,1,1,1]

V. RESULTS

The results obtained from detailed analysis are shown in Table 2, Table 3, and Table 4. Table 2 compares the cost obtained by solving [1] problem as shown in Table 1 using Hybrid PSO and GA [4] algorithms. Table 3 compares the (i) relative deviation from tighter lower bound (TCL), (ii) improvement achieved through Hybrid PSO algorithm over results obtained through GA algorithm [4], (iii) efficiency in terms of execution time taken by Hybrid PSO algorithm. Table 4 compares the detailed solution found by Hybrid PSO with GA solution [4].

Table 2 shows that 71% of Hybrid PSO solutions are either better or similar to best result obtained from GA algorithm, while only 41% of GA solution are better or similar to best result obtained from Hybrid PSO algorithm. So, in majority of cases Hybrid PSO performed better than GA algorithm.

Table 3 shows that average relative deviation from TCL is 19.805% using Hybrid PSO and worst average relative deviation from TCL is 21.261% using GA algorithm, average improvement over GA is 0.962% using PSO, and average CPU utilization time is 14.049 sec using PSO. It is also important to note that GA differs with PSO algorithm for high utilization as well as low utilization cases. GA found worst relative deviation from TCL for higher utilization but results for lower utilization cases are comparatively closed to PSO algorithm.

Table 4 shows the detail comparison of values for T and k_i(i.e., i=1,2,…10) using Hybrid PSO with GA algorithm. For low utilization cases 50 to 92 k_i have different values but for high utilization cases 95 to 99 all k_i have same value '1'. Hybrid PSO found same value for T and $k_i$which gives low deviation from TCL. GA found the same value for k_i but failed to found value for T similar to Hybrid PSO algorithm and therefore it results in high deviation from TCL.

VI. CONCLUSION

This research presented hybridization scheme based on Particle Swarm Optimization and Golden Section Search to solve the ELSP problem under basic period approach. This Hybrid PSO technique used PSO optimization to find the optimum value of k_i's and followed by GSS to find the basic period T. The feasibility of the solution is guaranteed with a constraint that ensures the items assigned in each period can be produced within the length of the period. The experimental results indicate following outcomes:

- The hybridization scheme was able to find comparatively better BP solutions than GA [4]for the low utilization problems.
- The hybridization scheme was also able to find comparatively better BP solutions than GA [4] for the high utilization problems.

References

[1] E. Bomberger,"A dynamic programming approach to a lot size scheduling problem," Management Science, vol. 12, no. 11,pp. 778–784, July 1966.

[2] S. E. Elmaghraby, "The Economic Lot Scheduling Problem (ELSP): Review and Extensions," in Management Science, vol. 24, no. 6, pp. 587–598, February 1978.

[3] S. E.Elmaghraby, "An Extended Basic Period Approach to the Economic Lot Scheduling Problem (ELSP)," Production and Industrial Systems: Future Development and the Role of Industrial and Production Engineering, Taylor and Francis, pp. 649–662.

[4] M. Khouja, Z. Michalewicz, and M. Wilmot, "The use of genetic algorithms to solve the economic lot size scheduling problem," European Journal of Operational Research, vol. 110, no. 3, pp. 509-524, November 1998.

[5] K. Thanushkodi and K. Deeba, "On performance analysis of hybrid algorithm (Improved PSO with Simulated Annealing) with GA, PSO for multiprocessor job scheduling," WSEAS Transaction on Computers, vol. 10, no. 9, pp. 287-300, September 2011.

[6] G. Dobson, "The Economic Lot-Scheduling Problem: Achieving Feasibility Using Time-Varying Lot Sizes," Operation Research, vol 35, no. 5,pp. 764-771, September 1987.

[7] F.Hanssmann, "Operations Research in Production and Inventory," John Wiley and Sons, pp. 158-160, 1962.

[8] J. Kennedy andR. C.Eberhard, "Swarm intelligence,"Morgan Kaufmann Publishers,2001.

[9] H. Aytug, M. Khouja, and F. E.Vergara,"Use of genetic algorithm to solve production and operations management problems: A review," International Journal of Production Research, vol. 41, no. 17, pp. 3955–4009, November 2003.

[10] M. Ben-daya and M. Al-Fawzan, "A tabu search approach for the flow shop scheduling problem," European Journal of Operational Research, vol. 109, no. 1, pp. 88–95, August 1998.

[11] R. W. Eglese, "Simulated annealing: A tool for operational research," European Journal of Operational Research, vol. 46, no. 3, pp. 271–281, June 1990.

[12] L. Gaafar,"Applying genetic algorithms to dynamic lot sizing with batch ordering," Computers & Industrial Engineering, vol. 51, no. 3, pp. 433–444, November 2006.

[13] MathWorks, "Find minimum of single-variable function on fixed interval - MATLAB", http://www.mathworks.com/help/matlab/ref/fminbnd.html, September 2012.

[14] H. Sun, H. Huang, and W. Jaruphongsa, "A genetic algorithm for the economic lot scheduling problem under extended basic period and power-of-two policy," CIRP Journal of Manufacturing Science and Technology, vol. 2, no. 1, pp. 29-34, 2009.

[15] M. F. Tasgetiren, O. Bulut, and M. M. Fadiloglu, "A Discrete Harmony Search Algorithm for the Economic Lot Scheduling Problem with Power of Two Policy," IEEE World Congress on Computational Intelligence, June 2012.

[16] S. A. Raza and A. Akgunduz, "A comparative study of heuristic algorithms on Economic Lot Scheduling Problem," Computer & Industrial Engineering, vol. 55, no. 1, pp. 94-109, August 2008.

[17] R. Kubota and H. Tamukoh, "A modified particle swarm optimization considering component combined with personal best positions," Recent Researches in Circuits, Communications and Signal Processing, pp. 55-58, 2013.

[18] M. F. Tasgetiren, O. Bulut and M. M. Fadiloglu, "A discrete harmony search algorithm for the economic lot scheduling problem with power of two policy," IEEE World Congress on Computational Intelligence, 2012.