Balancing of assembly lines with collaborative robots

As the automation in manufacturing department is increasing significantly, several robots are developed to carry out several tasks to increase the production rate. However, not all tasks can be automated due to the lack of flexibility with available robotic technologies. Humans on the other hand are flexible, adaptable according to market demands, and also have decision making skills with creativity. To remain relevant, many manufacturing enterprises show interest in human robot collaboration where the skills of human and the speed, endurance, and accuracy of robots are brought together to increase production. This also helps the human workers as the repetitive and stressful tasks are done by robots.

In this paper, the authors propose mixed-integer-programming formulation for balancing and scheduling of assembly lines with collaborative robots(cobots). This model helps in both assignment of the cobots to different stations with balanced distribution of workload to both human workers and the robotic partners. The main aim of this paper is to minimize the cycle time with given number of stations (ALP II). To do this they have implemented hybrid genetic algorithm.

The hybrid genetic algorithm used in this paper has these seven steps,

1. Population generation
2. Fitness evaluation and improvement
3. Parents selected randomly
4. Crossover (one point crossover using 2 parents)
5. Mutation
6. Fitness evaluation of off-springs generated and improvement
7. Greedy selection (survival of the fittest)

The improvement phase that is used alongside calculation of the fitness evaluation is used to improve the solution set. This helps in balancing the cycle time of each station and distributes the work load accordingly. The stations with robots assigned have more task load than the one with no robots.

The results according to the paper say that, productivity increases by deploying cobots effectively. Since this was a novel approach, there were no models or algorithms to bench mark their results. Due to the OOP approach, further development of genetic algorithms in optimizing multi objective problems with human robot collaboration is promising.

MOO USING PSO (A Multi-Objective Particle Swarm Optimization Algorithm Based on Gaussian Mutation and an Improved Learning Strategy)

In all multi objective optimization(MOO) problems, the 2 main goals are to have good convergence and diversity in the population. Also MOO problems itself is difficult to solve as unlike the single objective problems this has many solutions know as pareto optimal set. Over the decade, many have modified PSO to solve these kind of challenging problems. In this paper, a novel MOIPSO(multi objective improved particle swarm optimization) is introduced which is based on Guassian mutation and improved learning strategy.

The improvements made by this paper on PSO are,

1. Elitist reservation mechanism – in this they provide an external archive(EA) to memorize all the non-dominated solutions(pareto optimal solutions in the population). This EA gets updated on each iteration as this changes for new population(generations). This EA has limited storage space and if this is exceeded in any iteration, then according to the crowding entropy, the solutions that are crowded more are removed from the EA.
2. Guassian Mutation Strategy – this paper introduces new guassan mutation throw point strategy, which throws points randomly into EA and current population. This throwing of points is done according to the crowding entropy, i.e,,. Points are thrown in between the sparse solutions from the EA and the population.
3. Improved learning strategy – The formula to update the velocity and to find global best, a new method is proposed here which uses EA and current population to have good convergence.
4. Update External Archive(EA) – a set of rules are used to update EA which considers the current population, thrown points, off-springs and old EA.
5. Population Elitist incremental strategy – to increase the convergence, the offspring generation used both EA and parent influence.

The performance measures used to benchmark this proposed method are, Convergence Measure Indicator and Distribution Measure Indicator (uses Distribution uniformity and Distribution width).

This algorithm is benchmarked with three algorithms namely, non-dominated sorting genetic algorithm(NSGA-II) and multi objective particle swarm optimization(MOPSO). From the results it shows that it performed better than these two. The future scope for this method is to optimize the parameters it used to get better solutions with less iterations.

The drawbacks of this method are, If number of throwing points are high, this increases the running time, and if the pareto optimal front is complicated rather than a smooth curve, then Guassian throw point strategy will fail.

MOO approaches USING PSO (Multi-Objective Particles Swarm Optimization Approaches)

This paper elaborates on the different approaches used to solve multi objective optimization problems using particle swarm optimization(swarm intelligence based optimization). PSO has attracted the interest of many people due to its simplicity to use for a naïve user, its efficiency and effectiveness on solving single objective problems.

In solving multi objective problem, the major difficulty that many face is that, it has many different solutions as the objective functions themselves have many tradeoffs. So, unlike single objective where only one solution is found, here many solution set is found named pareto optimal set. Many evolutionary algorithms(EA) showed promising results in solving these MOO problems. As PSO shares many similarities with EA, PSO was opted to solve MOO problems. Many modifications were made to PSO to solve MOO and this paper, elaborates the different approaches used by different authors.

The main goal of solving MOO problems is to find as many solutions as possible that lies in the pareto front. So, the population must be diverse and should have better convergence towards the pareto front.

In general, PSO uses a velocity vector for each particle to update its position in the swarm. This updation is affected by its own personal best and the swarm best. This helps PSO algorithms to exploit the search space. Many different types of updating the position of he particles are demonstrated in the paper. The PSO approaches to solve MOO is categorized into two methods namely,

1. Approaches that evaluate the each objective function sepearately
2. Pareto based schemes (uses external archive(EA))

This paper divided the PSO approaches and categorized them as,

1. Algorithms that exploit each objective function separately – These approaches includes the PSO techniques that combine the MO to SO or use each objective function in turn to update the swarm. It also includes various techniques used to aggregate the objective functions and function ordering.
2. Non-pareto vector evaluated approaches – This is similar to vector evaluated genetic algorithms(VE-GA). In this VE-PSO, many swarms were created with each swarm devoted to on objective function. Also, later development of this technique and the modifications made are also explained with several different methods to update the swarms and make it converge to the pareto front.
3. Algorithms based on Pareto Dominance – These algorithms uses the pareto dominance sorting to find the leader of the swarm that guides the other particles in it to the pareto front. Many approaches such as MOPSO that uses this technique were explained. These approaches uses EA to store the dominant solutions so as to select the leader from them. To avoid converging to a single solution, many methods such as crowding factor, crowding density were used to avoid them. Also algorithms that do not use EA are also explained(AMOPSO). To better optimize these, rather than using singe EA, many EA were used to store particles best and global best.

This paper also describes the future developments that are promising, such as non pareto algorithms as they have lesser computational burden compared to other developed algorithms recently. However, the non-pareto based algorithms are efficient and simple to implement and gain more interest in many manufacturers. Parallel implementation of MOO PSO approaches are also an active research area.