ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com
International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE) Vol. 2, Special Issue 12, April 2016.

Particle Swarm Optimization Approach for Flow Shop Scheduling Problem – A Case Study

Dr.S.Sridhar
Assistant professor, Department of Mechanical Engineering
PSNA College of Engineering and Technology
Dindigul, Tamilnadu
sri_2855@yahoo.com.

S.Ashok kumar, T.Aravindhan
B.E (Final Year)., Department of Mechanical Engineering
PSNA College of Engineering and Technology
Dindigul, Tamilnadu
ashoksounder1995@gmail.com.

Abstract— This paper considers a flow shop scheduling problem. The flow shop scheduling is characterized "n" jobs and "m" machines in series with unidirectional flow of work with variety of jobs being processed sequentially in a single pass manner. Most real world problems are NP-hard in nature. The essential complexity of the problem necessitates the use of meta-heuristics for solving flow shop scheduling problem. The paper addresses the flow shop scheduling problem to minimize the makespan time with specific batch size using Particle Swarm Optimization(PSO) and comparing the results with an real time company production sequence.

Keywords— flow shop, makespan time, meta-heuristics, PSO algorithm,industrial company sequence.

I. INTRODUCTION

Scheduling plays vital role in manufacturing and service industries. Effective scheduling techniques should be required for improving the efficiency of industries. This paper considers flow shop scheduling problems to minimization of makespan time with a specified batch size. The flow shop is characterized of 'n' jobs 'm' machines. Each job has to be processed first at machine 1, then machine 2 and so on. The scheduling problem in a flow shop scheduling problem using PSO has been the subject of considerable research. The flow shop scheduling model was first developed by Johnson (1954). Johnson developed an exact algorithm to minimize the makespan for 2- machines flow shop scheduling problems. The flow shop scheduling problem has been proved to be NP hard. Due to the complexity of the problem, it is difficult to develop exact methods to solve this problem. Hence, researchers proposed different heuristics and meta-heuristics to solve the flow shop scheduling problems. The important heuristics were developed by Rajendran and Chaudhri (1990) and also proposed to solve the flow shop scheduling problems. A greedy heuristic algorithm was addressed by Baraz and Mosheiov to minimize the makespan for flow shop scheduling problems. Similarly a Solar Panel holder with clamp (SPHC) was investigated by team members in a clamp manufacturing company. The system analysed is composed of seven machines used for shearing, punching, and spot welding, grinding, drilling and painting in series. The number of machines in shearing, punching, and spot welding, grinding, and drilling stages are equal respectively (say one machine each). All jobs are performed one by one machine in a unidirectional flow.

A. Objective Function

A flow shop scheduling is characterized by unidirectional flow of work with a variety of jobs being processed sequentially in a one-pass manner. A flow shop is which 'n' jobs to be processed through 'm' machines environment. The processing times of all the jobs are well known in advance and all the jobs have been processed in the same order in various machines. A particular set of jobs can be sequenced through all the machines and each sequence will have an objective function as makespan time with the specified batch size. It is difficult to suggest a sequence, which will optimize the makespan time. In this paper, proposed the PSO algorithm which will optimize the sequence so as to achieve minimum value of makespan time with the specified batch size using n number of iterations. More the iterations bring more the quality of the results.

$$C_{max} \geq C_{im} \; for \; all \; i = 1,2,3 \dots \dots . n \qquad Equ\ (1)$$

Notations

C_{max} – Minimization of Makespan Time
C_{im} – Completion time of the job (i) on Machine (m)

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com
International Journal of Advanced Research in Management, Architecture, Technology and Engineering
(IJARMATE) Vol. 2, Special Issue 12, April 2016.

n – Number of jobs

B. Illustration of NPS

A Gantt chart developed for generating Permutation Schedules (PS) can yield solutions of good quality in a flow shop scheduling problems. But the solutions may not be satisfactory. Because, the job has to follow a fixed operation sequence at each machine even though there is required operation for the job at all machines. Therefore, a better schedule performance can usually be obtained by allowing jobs to change the operation sequence at different machines like Non Permutation Schedule (NPS).

II. LITERATURE SURVEY

During the last three decades many research works have been done in this area. The flow shop problems are said to problem is NP-hard. The solutions for these problems are obtained by heuristics or meta-heuristics approach. Many heuristic algorithms have been developed to generate optimum schedule and part releasing policies. Most of these algorithms include enumerative procedure, mathematical programming and approximation techniques viz. Linear programming, integer programming, goal programming, dynamic programming, transportation and network analysis, branch and bound, lagrangian relaxation, priority rule-based heuristics, local search algorithms [Iterative search (ITS), Tabu search (TS), Threshold Accepting (TA), Simulated Annealing (SA)], Evolutionary Genetic algorithm (GA) etc. Among these techniques few are specific to particular objective like makespan time and tardiness etc., few are specific to particular problem instances with respect to computational time needed.
Johnson (1954) is believed to be the first who introduced flow shop scheduling. Since then, flow shop scheduling has become one of the most interesting topics among researchers and practitioners these are different forms of flow shop optimization such as minimization of the makespan which is one of the most popular one. Turner and Booth (1987) compared two famous heuristics with a set of 350 random problems. Ponnambalam et al. (2001) compared five different heuristics against only 21 typical test problems. Ruiz and Maroto (2005) presented a review and comparative evaluation of heuristics and meta-heuristics for the permutation flow shop problem with the makespan criterion. They compared 25 methods, ranging from the classical Johnson’s algorithm to the most recent meta-heuristics. Lian et al. (2006) applied an efficient similar particle swarm optimization algorithm (SPSOA) to the PFSS problem with the objective of minimizing the makespan. Tasgetiren et al. (2007) solved the permutation flow shop sequencing problem (PFSP) with a particle swarm optimization algorithm (PSO). They considered the objectives of minimizing makespan and the total flow time of jobs. Ruiz and Stutzle (2007) presented a new iterated greedy algorithm that applies two phases iteratively, named destruction, where some jobs are eliminated from the incumbent solution, and construction, where the eliminated jobs are reinserted into the sequence using the well known NEH construction heuristic. Naderi and Ruiz (2010) studied a new generalization of the regular permutation flow shop scheduling problem (PFSP) referred to as the distributed permutation flow shop scheduling problem or DPFSP. Under this generalization, they assumed that there are a total of F identical factories or shops, each one with m machines disposed in series. A set of n available jobs have to be distributed among the F factories and then a processing sequence has to be derived for the jobs assigned to each factory. Their optimization criterion was the minimization of the maximum completion time or makespan among the factories. Dong et al. (2009) presented an integrated local search algorithm to solve the permutation flow shop sequencing problem with total flow time criterion. They showed the effectiveness and superiority of their method over three constructive heuristics, three ant-colony algorithms and a particle swarm optimization algorithm. Vallada and Ruiz (2009) worked on a cooperative meta-heuristic method for the permutation flow shop scheduling problem considering two objectives separately: total tardiness and makespan. They adopted the island model where each island runs an instance of the method and communications begin when the islands are reached to a certain level of evolution. Farahmand Rad et al. (2009) showed five new methods that outperform the well-known NEH heuristic as supported by careful statistical analyses using the well-known instances of Taillard. The proposed methods attempt to counter the excessive greediness of NEH by carrying out re-insertions of already inserted jobs at some points in the construction of the solution. Vallada and Ruiz (2010) presented three genetic algorithms for the permutation flow shop scheduling problem with total tardiness minimization criterion. The algorithms include advanced techniques like path re-linking, local search and a procedure to control the diversity of the population. Sridhar et al (2010) & (2014) investigated stainless steel industry and Optimised company production sequence in hybrid flow shop using simulated annealing algorithm.

In this paper, we consider a permutation flow shop scheduling problem with the objective of minimizing the makespan. The method is then solved using a particle swarm optimization (PSO) algorithm and it is employed to solve the problem. PSO algorithm, developed by Kennedy and J. Eberheart (1995)and it was first intended for simulating social behaviour as a stylized representation of the movement of organisams in a bird flock or fish school.it is an computational method that optimises the problem by iteratively trying to improve a candidate solution with regard to given measure of

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com
IJARMATE International Journal of Advanced Research in Management, Architecture, Technology and Engineering
(IJARMATE) Vol. 2, Special Issue 12, April 2016.

quality. It solves a problem by having a population of candidate solutions, here dubbed particles and moving these particles around in the search-space according to simple mathematical formulae over the particle position and velocity.

III. CASE STUDY

R.N Solar Energies is one of the groups of R.N. Steels and Engineering (P) Ltd., located at Dindigul. Their main product is supporting beams and holding clamps used to carry the photovoltaic cell structured panels. The product concerned in this work is Clamps and Supporting beams. The problem considered in this work is optimum sequencing & scheduling for 500 numbers of clamps and beams. The main objective is to minimize the makespan time.

The present work investigates a production system in a Manufacturing company that manufactures various clamps and beam components. The system is composed of five workstations (stages), shearing, filing, punching, drilling, grinding, spot welding, and painting.

At the first workstation, the shearing machine is used to perform cutting operation and followed by filing operation with help of hand grinder or manual rough file the sharp edges are blunt in order to avoid handling damages. At the next workstation, the punching machine is used to perform punching operation, producing punch in required marking to exact location of drilling tool. The next workstation to be drilling to drill through holes in the work piece and followed by grinding operation to get a smooth surface finish performed by a surface grinder. In the next workstation welding operation is performed by spot welding machine. The last work station is to paint the jobs by a spray painting machine.

Each workstation consists of equal number of machines. All jobs are performed one by one in all machines as said earlier the job flow is unidirectional for flow shop problems. In this scenario, this work proposes research on optimisation of sequencing and scheduling in flow shop environment with the makespan objective.

A. Number of Machines

The details of various types of machines and jobs which has been produced in the R.N Solar Energies Company for manufacture of clamps and supporting beams are given in Table 1 and Table 2

Table 1: Number of machines

Sl.no.	*Particulars*	*Number of machines*
1	Shearing machines	1
2	Filing machines	1
3	Punching machines	1
4	Drilling machines	1
5	Grinding machines	1
6	Spot welding machines	1
7	Spray painting kit	1

B. Data for Clamp and Supporting Beam

The details regarding various components, operations, machines, quantity, machining time and batch size for clamp and supporting beam production, are given in Table 2.

Table 2: Data for production component

Component	*Operation*	*Avg. time (sec)*	*Batch size*
Base plate	shearing	16	100
Base plate	filing	10	100
Base plate	punching	13	100
Base plate	drilling	45	100
Base plate	grinding	42	100
Base plate	spot welding	14	100
Base plate	painting	18	100
Stiffener L&R	shearing	19	100
Stiffener L&R	filing	06	100
Stiffener L&R	punching	10	100
Stiffener L&R	drilling	21	100
Stiffener L&R	grinding	36	100
Stiffener L&R	spot welding	16	100
Stiffener L&R	painting	19	100
Centre piece	shearing	13	100
Centre piece	filing	05	100
Centre piece	punching	08	100
Centre piece	drilling	22	100
Centre piece	grinding	21	100
Centre piece	spot welding	09	100
Centre piece	painting	13	100
Supporting beam	shearing	31	100
Supporting beam	filing	36	100
Supporting beam	punching	11	100
Supporting beam	drilling	26	100
Supporting beam	grinding	53	100
Supporting beam	painting	42	100

Mc's\jobs	J1	J2	J3	J4	J5
M1	16	19	13	19	31
M2	10	06	05	06	36
M3	13	10	08	10	11
M4	45	21	22	21	26
M5	42	36	21	36	53
M6	14	16	09	16	0
M7	18	19	13	19	42

IV. PARTICLE SWARM OPTIMIZATION

PSO is an evolutionary computation method developed by Kennedy & Eberhart (1995). It stimulates the social behaviour of bird flocking or fish schooling. Like other non-traditional techniques, the PSO is also a population based optimization technique. It delineates a type of biological social system, which depicts the collective behaviour of simple

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com
IJARMATE
International Journal of Advanced Research in Management, Architecture, Technology and Engineering (IJARMATE) Vol. 2, Special Issue 12, April 2016.

individuals interacting with their environment and one another. It is inspired by the movement and intelligence of swarms. A swarm is a structured collection of interacting organisms such as bees, ants or birds. Each organism in a swarm is a particle or an agent. Particles and swarms in PSO represent the individuals and populations as in other evolutionary algorithms.

The initiation of the PSO algorithm is done with a population of random solutions, denoted as random particles and then searches for optima by updating generations. New generations are formed by updating velocity. The potential solutions are called particles. Each particle is updated by following two "best" values in each iterations as p_{best} and g_{best}. The particles fly through the multi-dimensional search space and follow the current optimum particles. Each particle has particular velocity, with which the particles are carried to new positions and are evaluated for fitness values according to their positions. PSO does not combine the survival of the fittest whereas all other evolutionary algorithms do. Since each particle exchanges its information with the particles in the neighbourhood, after some number of iterations, the swarm loses its diversity and the algorithm converges to the optimal solution. All particles in the pool are kept during the whole run. The PSO is carried out for optimal value of the required number of iterations. The good solution is reached among the updated generations. PSO is used as an approach that can be used across a wide range of applications, which include function optimization, artificial neural network, fuzzy system control, as well as for specific applications focused on a special requirement.

A. Basic Elements of PSO

The basic elements of PSO algorithm are particle, population, permutation, particle velocity, personal best, global best (and) termination criterion.

1. Particle

Xi denotes the i^{th} particle in the swarm at iteration t and is represented by n number of dimensions given as Equation

$$[X_i]^t = [(x_{i1})^t, (x_{i2})^t, \dots, (x_{in})^t]$$

Where, $(x_{ij})^t$ is the position value of the i^{th} particle with respect to the j^{th} dimension $(j= 1,2,\dots, n)$.

2. Population

pop^t is the set of ρ particles in the swarm at iteration t denoted in Equation

$$pop^t = [(X_1)^t,(X_2)^t,\dots\dots,(X_\rho)^t]$$

3. Permutation

It introduces a new variable $(\pi_i)^t$ which is a permutation of jobs implied by the particle$(X_1)^t$. It can be described in Equation

$$[\pi_i] = [(\pi_{i1})^t,(\pi_{i2})^t)\dots\dots(\pi_{in})^t)] \quad (3.3)$$

Where, $(\pi_{ij})^t$ is the assignment of job j of the particle i in the permutation at iteration t.

4. Particle velocity

$[V_i]^t$ is the velocity of particle i at iteration t. It can be defined as Equation

$$[V_i]^t = [(v_{i1})^{t,}(v_{i2})^t\dots\dots(v_{in})^t]$$

Where, $(v_{ij})^t$ is the velocity of particle i at iteration t with respect to the j^{th} dimension.

5. Personal best

The $[P_i]^t$ represents the best position of the particle i with the best fitness until iteration t, so the best position associated with the best fitness value of the particle i obtained so far is called the personal best. For each particle in the swarm, $(P_i)^t$ can be determined and updated at each iteration t. In a minimization problem with objective function $f(\pi_i)^t$ where $\pi_i)t$ is the corresponding permutation of particle $(X_i)^t$, the personal best $(P_i)^t$ of the i^{th} particle is obtained in such a manner that $f(\pi_i)^t \leq f(\pi_i)^{t-1}$ where $\pi_i)^t$ is the corresponding permutation of personal best $(Pi)^t$ and $\pi_i)^{t-1}$ is the corresponding permutation of personal best $(P_i)t-1$. To simplify, we denote the fitness function of the personal best as $(f_i)^{pb} = f(\pi_{i1})^t$. For each particle, the personal best is described as Equation

$$[P_i]^t = [(p_{i1})^t,(p_i\ )^t\dots\dots(p_{in})^t]$$

Where, $(p_{ij})^t$ is the position value of the i^{th} personal best with respect dimension $(j = 1,2,\dots, n)$.

6. Global best

The $[G]^t$ denotes the best position of the globally best particle achieved so far in the whole swarm. For this reason, the global best can be obtained such that $f(\pi)^t \leq f(\pi_i)^t$ for $i = 1,2,\dots, \pi$ where $(\pi)^t$ is the corresponding permutation of lobal best $G\ t$ and $(\pi_i)^t$ is the corresponding permutation of personal best $[P_i]^t$. To simplify, we denote the fitness function of the global best as $(f)^{gb} = f(\pi)^t$. The global best is then defined as Equation

$$[G]^t = [(g_1)^t,(g_2)^t\dots\dots(g_n)^t]$$

Where, $(g_j)^t$ is the position value of the global best with respect to the j^{th} dimension $(j = 1,2,\dots, n)$.

7. Termination criterion

It is a condition that the search process will be terminated. It might be maximum number of iteration or maximum CPU time to terminate the search.

B. Step by Step Procedure for PSO

ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com
IJARMATE International Journal of Advanced Research in Management, Architecture, Technology and Engineering
(IJARMATE) Vol. 2, Special Issue 12, April 2016.

The basic step by step procedure of the PSO algorithm given is as follows,

Step 1: Initialization

Initialize a population of n particles randomly.

Step 2: Calculate the fitness function

Calculate the fitness value for each particle, if the fitness value is better than the best fitness value in history $(p_{ij}^{\,t-1})$.Then set current value as the new p^{best}.

Step 3: Choose the best fitness value

Choose particle with the best fitness value of all the particles as the g^{best}. $(g_j)^{t-1}$

Step 4: Calculate the particle velocity and position

For each particle, calculate velocity and position by using the equation,

$$[V_{ij}]^t = [(v_{ij})^{t-1} + c_1 r_1\{(p_{ij})^{t-1} - (x_{ij})^{t-1}\} + c_2 r_2 \{(g_j)^{t-1} - (x_{ij})^{t-1}\}]$$

$$[X_{ij}]^t = (x_{ij})^{t-1} + (v_{ij})^t$$

Where,

$(v_{ij})^{t-1}$ = Velocity of particle i at t-1th iteration
$(V_{ij})^t$ = Velocity of particle i at tth iteration
$(x_{ij})^{t-1}$ = Position of particle i at t-1th iteration
$(X_{ij})^t$ = Position of particle i at tth iteration
c_1 = Acceleration factor related to p^{best}
c_2 = Acceleration factor related to g^{best}
r_1 = Random number between 0 and 1
r_2 = Random number between 0 and 1
$(g_j)^{t-1}$ = global best position of swarm
$(p_{ij})^{t-1}$ = local best position of particle

Step 5: Update the particle velocity and position

Each particle velocity and position is updated according to the dimensions.

Step 6: Termination of PSO

Terminate if 750 iterations is reached. Otherwise, go to Step 2.

There is a communication between the each particle delivers its information with others. A particle exchanges its information with the particles in the neighbourhood. Therefore, after some number of Iterations the swarm loses its diversity and the algorithm converges to the optimal solution. Since PSO consists of simple concepts and mathematical operations with little memory requirements it is fast and appealing in use for many optimization problems. To verify the PSO algorithm, comparisons with simulated annealing algorithm is made. Computational results show that the PSO algorithm is very competitive. Computational results show that the local search can be really guided by PSO.

V. RESULTS

A. *Computational Analysis*

The PSO is used to find optimum or near optimum sequence for the problem given in the Table. The makespan time for the best sequence obtained using PSO procedure is compared with the makespan time for the sequence that the company currently using. The comparison reveals that the makespan time of PSO procedure gives better solution than the makespan time of company sequence and corresponding comparison is shown in

Table 3: Comparison of company result and PSO result

S.No	PARTICULARS	Company result	PSO RESULT
1	Makespan time(sec)	19223	18878
2	Sequence	12345	42513

B. *Results Comparison*

- The proposed PSO is applied to find optimum or near optimum sequence for the problem given in the Table 3.
- The makespan time for the best sequence obtained using PSO procedure is compared with the makespan time for the sequence that the company currently using.
- Table represents the comparison of makespan time using PSO with that of company existing production sequence.
- The result shows that PSO is capable of providing better solution than the existing production sequence of the company.
- Influence of job size on execution time is significant.

C. *Limitations of Research Work and Future Scope*

- The scheduling problems considered in this work are of n jobs m machines. In this research work is single objective have been considered. The objective like multi objective function may be considered.
- Tool set up time and job setup time are considered as zero. They may be considered in the future work.
- Machine break down time is not considered. In future, the problems may be considered with the machine breakdown.
- The solution may be obtained using other heuristic approaches and comparison may be made with the current solution.

IJARMATE
Your ult-MATE Research Paper !!!
ISSN (ONLINE): 2454-9762
ISSN (PRINT): 2454-9762
Available online at www.ijarmate.com
International Journal of Advanced Research in Management, Architecture, Technology and Engineering
(IJARMATE) Vol. 2, Special Issue 12, April 2016.

VI. CONCLUSION

This work addresses a flow shop scheduling environment that manufactures the product clamp and supporting beam.The aim is to determine optimal or near optimal schedule for 'n' jobs, which processed at'm' machines. A PSO algorithm is proposed for get the optimum or near optimum schedule and sequence. The makespan time for the best sequence obtained using PSO procedure is compared with the makespan time for the sequence that the company currently using. The comparison reveals that the PSO is capable of providing better solution than existing production sequence. So it is concluded that the PSO proposed for the problem under consideration can very well be applied to find better schedule.

VII. REFERENCES

1.Kennedy.J and Eberheart.R,(1995) "Particle Swarm Optimization", *Proceedings Of IEEE Intenational Conference On Neural Networks* pp.1942-1948

2. Shi and Eberheart.R,(1995) "a modified Particle Swarm Optimizer", *Proceedings Of IEEE Intenational Conference On Evolutionary computation.* pp . 66-73

Kennedy.J and Eberheart.R,(1995) "Particle Swarm Optimization", *Proceedings Of IEEE Intenational Conference On Neural Networks* pp.1942-1948

3. S. M. Johnson, "Optimal two- and threestage production schedules with setup times included", *Naval Research Logistics Quarterly*, Vol. 1, No. 1, pp. 61 – 68, 1954.

4. C. Rajendran and D. Chaudhri, "Heuristic algorithm for continuous flow-shop problem", *Naval Research Logistics*, Vol. 37, No. 5, pp. 695 – 705, 1990.

5. M. Saravanan and S.Sridhar, "Optimization of Realistic Multi-Stage Hybrid Flow Shop Scheduling Problems with Missing Operations Using Meta-Heuristics", *International Journal of Engineering and Technology (IJET)*, Vol. ISSN : 0975-4024 Vol 6 No 1 Feb-Mar 2014, pp. 467-479.

6. M. Saravanan, T. Prabaharan and S.Sridhar, "Optimisation of sequencing and scheduling in hybrid flow shop environment using heuristic approach", *Int. J. Logistics Economics and Globalisation*, Vol. 2, No. 4, pp. 331-351.

7.Mohammad Kazem Sayadi, Reza Ramezanian, Nader Ghaffari-Nasab, "A discrete firefly meta-heuristic with local search for makespan minimization in permutation flow shop scheduling problems", *International journal of Industrial Engineering computations*, pp.1-10, 2010.

8. Vallada, E. & Ruiz ,R. "Co-operative Metaheuristics For The Permutation Flowshop Sheduling Problems", *European journal of Operational Research,* 193, pp-365-376, (2009)

9. Ponnambalam, S.G., Aravindan, P., & Chandrasekaran ,S. "Constructive and Improvement Flowshop Scheduling Heuristics- An Extensive Evolution", *Production Planning And Control,* 12(4), pp-335-344, (2001)

10. Farahmand Rad, S., Ruiz, R., & Boroojerdian ,N. "New High Performing Heuristics For Minimizing Makespan in Permutation Flowshops", *Omega*, 37, pp-331-345, (2009)

11. Naderi, B. & Ruiz ,R. "The Distributed Permutation Flowshop Sheduling Problem", *Computers and Operation Research,* 37, pp-754-7688, (2010)

12. Ruiz ,R. & Maroto, C. "A Comprehensive review and Evaluation of Permutation Flowshop Heuristics", *European journal of Operational Research,* 165, pp-479-494, (2005)

13. Lian ,Z., Gu, X., & jiao, B. "A similar particle swarm optimization algorithm for permutation flow shop scheduling to minimize makespan", *Applied Mathematics and Computation,* 175, pp-773-785, (2005)

14. Ruiz ,R. & Stutzle, T. "A simple effective iterated greedy algorithm for permutation flowsop sheduling problem", *European journal of Operational Research*, 177, pp-2033-2049, (2007)