DEVELOPMENT OF EVOLUTIONARY ALGORITHM FORMULTI-CRITERIA OPTIMIZATION IN ASSEMBLY BASEDSHOP

Aravind yuvaraj1,K.Ganesan2, Balamurugan3
Assistant Professor, Department of Mechanical Engineering
Dhanalakshmi Srinivasan College of Engineering and Technology, Mamallapuram

ABSTRACT:

Effective inventory management is a crucial aspect of a successful business practice. Theinventorymanagementisneededasbeingaportionofsupplychainnetworktoguardthemanufacturing program towards any type of disturbance. Moreover, it also prevents the system fromworking outofstock,componentsandproducts

Inventory mismanagement can be detrimental to a business, especially considering the weight theseitems carry. Inventories that run out of control can lead to significant losses that the company may notbe able to recoup. Considerable investment is required to develop adequate stock. Poorly managedsuppliesleadtoprofitloss.

In this paper it is proposed to carry out the development of multi criteria algorithm for a specific casestudy. These heuristicsare testedandinferenceswillbe made.

KeyWords—**EvolutionaryAlgorithm,Multi-criteria, optimisation**

INTRODUCTION:

Effectiveinventorymanagementisacrucial aspect of a successful business practice.The inventory management is needed as being aportionofsupplychainnetworktoguardthemanufacturingprogramtowardsanytypeofdisturbance.Moreover, italsopreventsthesystem from working out of stock, componentsand products.

Inventory mismanagement can be detrimental toabusiness,especiallyconsideringtheweighttheseitemscarry.Inventoriesthatrunoutofcontrolcanleadtosignificantlossesthatthecompanymaynotbeabletorecoup.Considerable investment is required to developadequate stock. Poorly managed supplies lead toprofitloss.

ForreducingthelossesduetoInventorymismanagementlikehavingmorestockthannecessary will cause the locking of cash or lessstockthanrequiredwillcauseproductionstoppage in Assembly lines which causes loss inbusiness.Hencecaretobetakenwhileplanningtheproductflowlinestomatchtheassemblyrequirements.Multi-criteriaoptimization (or multi-criteriaprogramming), also known as multi-objective or multi-attribute optimization,istheprocessofsimultaneouslyoptimizingtwoormoreconflictingobjectivessubjecttocertainconstraints.

Multicriteriaoptimizationproblemscanbefoundinvariousfields:productandprocessdesign, finance, aircraft design, the oil and gasindustry,automobiledesign,orwhereveroptimaldecisionsneedtobetakeninthepresenceoftrade-offsbetweentwoormoreconflictingobjectives.Maximizingprofitandminimizing the cost of a product; maximizingperformanceand minimizingfuelconsumptionofavehicle;andminimizingweightwhilemaximizingthestrengthofaparticularc omponentareexamplesofmulti-criteriaoptimization problem In this paper we used Evolutionary algorithm foroptimizing the inventory and thus increasing theprofit.

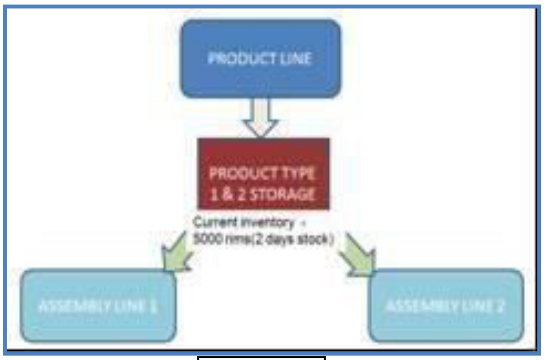
METHODOLOGY:

Methodologyofoptimizingtheinventorylevels between product shop and two differentassembly shops includes the data collection ofcycle times of the various processes involved intheproduct lineandtwoassembly lines .Thenthe product line and assembly lines are simulatedbymodelingtheshopfloorusingtheExtendsoftware.

The model will be simulated for validation andresults are verified with the results on the shopfloor.ThemaximumprofitfunctionwillbedevelopedandthevalidatedmodelwillbeoptimizedusingtheEv olutionaryoptimizerfunctionintheExtendsoftwareagainsttheobjective function. The results are compared andvalidatedbyrunningthemaximizefunction.
Based on the results of the maximizefunction an algorithm will be generated based ontheGeneticalgorithmusing“C”programming.

PROBLEMDEFINITION:

The problem includes a product shop whichproduces two different types of products whichareassembledintwodifferentassembliesasshowninFig.1below



e

Afterproduction ofcomponet ntsintheproduct saresto
demand. This paper aims at for optimizing theinventory in the storage area which will allowrunning the assembly line 1&2 without stoppagedue to want of products using the EvolutionaryOptimization approach

---

SIMULATIONMODELLING:

ModellingandSimulationisadisciplinefordevelopingalevelofunderstandingoftheinteraction of the parts of a system, and of thesystem as a whole. The level of understandingwhich may be developed via this discipline isseldomachievableviaanyotherdiscipline.

A simulation generally refers to a computerizedversion of the model which is run over time tostudytheimplicationsofthedefinedinteractions.Simulationsaregenerally iterativeintheredevelopment.Onedevelopsamodel,simulates it, learns from the simulation, revisesthe model, and continues the iterations until anadequatelevelofunderstandingisdeveloped.

Therevarioustypesofsimulationmodelingsoftwareareavailable.Someofthemare ExtendSimul8-simulationsoftware isaproductoftheSIMUL8Corporationusedforsimulating systems thatinvolveprocessingofdiscreteentities at discretetimes.

Flex sim - It is a discrete event manufacturingsimulationsoftware used in many fields such asManufacturing,Logistics,distribution,Transportation ,Oilfieldorminingprocess,networkingdata flowetc.

SimEvents - It is a continuous and discrete eventsimulationtooldevelopedbyMathWorks
Arena – It is discrete event simulation software,developedbySystemsModelling.Itusestheprocessorandsimulationlanguage.

ManufacturingStationusingExtend:
storage.Assemblylines1and2willpulltheproductsfromthestoragebasedonthecustomerSeveral identical machines have been gathered toperformsomemanufacturingoperation.Because of the variability in the interarrival times and thevariability of the processing times, a queue ofparts waiting processing forms in frontof themachines. The length of the queue changes overtime because of the randomness inherent in theprocess. Onegoalof an analysis is to measurethe average number in the queue and the averagetimedelayinthe queue.

The Arrival Process - Parts enter the station at aspecified arrival rate. The inverse of the arrivalrateisthemeantimebetweenarrivals.Variability in the arrival process means that thetime between arrivals at the station is a randomvariable.Amodeloftheprocessspecifiesaprobabilitydistributionforthetimebetweenarrivals.Thereareanumberofwell-knowndistributionsthatmightbeappropriatesuchasthe Normal or exponential distributions. The oneto use depends on the particular problem beingstudied and can be determined from analysis ofhistoricaldatausingstatisticaltechniques.

ServiceProcess–Astationinthesystemperforms some operation. Unless the station ishighlyautomated,theactualtimefortheoperationisprobably notconstant.Whenthistime is a random variable, a model must specifytheprobabilitydistributionforthetimeforreplicationoftheoperation.Asforthetimebetweenarrivalsthereareremanypossibledistributionsthatthisrandomvariablemighttake.

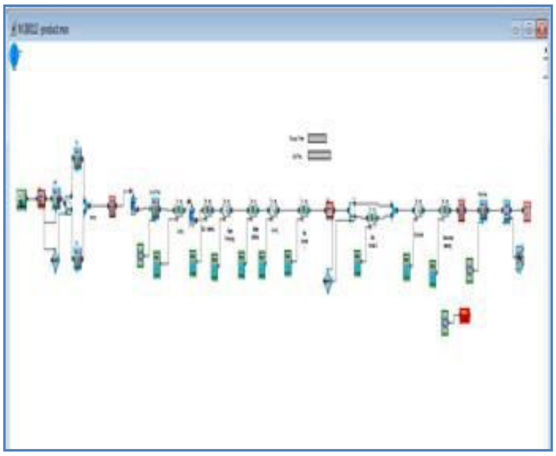
ExtendSimulation–Thesimulationisconstructed with various kinds of Extend blocksidentified by their graphical icons. Most blockshave parameters that are set by double clickingon the block. In the example, the blocks havebeenlabelledfordiscussionpurposes.

TheGeneratorblockcreatespartsthatpassthroughthesimulation.TheRandblockdeterminestheprobabilitydistributionoftheinterarrivaltimes.Inthiscasetheblockis labelledBetaDist.toindicatethattheinterarrivaltimeshavetheBetadistribution. Oncecreated,partspassfromtheGeneratorblockintotheFIFOQueue.TheActivity-Multiblock simulates the production activity.TheservicetimeisgovernedbytheRandblockconnectedtotheActivity.Theservicedistributionselectedfortheexampleistheexponentialdistributionwith mean 0.5.

After service completion, the part passes to theExit block. A simulation was setup to make 50separate runs, each run consisting of 100 events.An event is either an arrival or a service. ThenumbershownwithintheExitblockisthenumber of parts that left the system during thelast simulation run – 48. Since the total numberof events was 100, then 52 must have entered thesystem. When the simulation terminated, 4 partsremaininthe serviceactivityand queue.

Themodelprovidesaccesstosomesystemparameters in the fields at the upper right. Othercharacteristics are set by the parameters withintheblocks.SimulationcharacteristicsaresetwiththeSimulationSetupdialogintheRunmenu.

SimulationmodelofProductshopinExtend



`

Simulation model of the Assembly shop 1inExtend

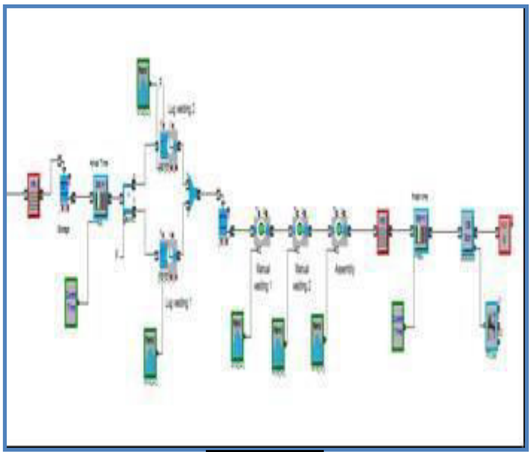


Fig.3

Simulationmodelof theAssembly line2

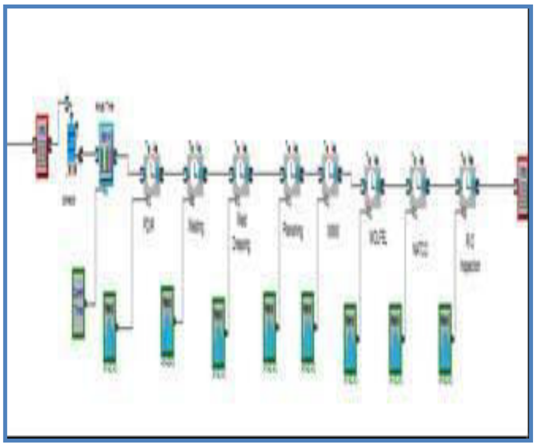


Fig.4

2. Evolutionaryoptimisationofthesimulationmodel

Optimization,sometimesknownas"Goalseeking," is a useful technique to automaticallyfindthebestanswertoaproblem.The ofmanuallytryingdifferentvalueswitheachmodelrun.

The downside to optimization is that the modelwill be run many times to find a value that mightbe easily found manually by the modeler. Thiscan take a long time with large models. Also,thereisasmallchancethatoptimizationwillconverge to a sub-optimum result. There are nooptimization algorithmsthatareguaranteed toconverge to the best answer in a finite time. Themoretimethatyoucangiveoptimizationtowork, the better chance that you will get the trueoptimumanswer toyourproblem.

Mostoptimizationalgorithmsthatcansolvestochasticmodels(modelswitharandomcomponent) use an initial population of possiblesolutions. Each solution is tried by running themodel several times, averaging the samples, andsorting the solutions. The best solution sets ofparameters are used to derive slightly differentsolutions that might be better. Each new derivedsolutioniscalledageneration.Thisprocesscontinues for enough generations until there areprobably no better solutions in sight, and then itterminates,setting up themodel with thebestonefound.

Theproblemwithalloptimizationalgorithmsstemfromtheinabilitytotellwhenthebestsolutionhasbeenfound,orevenifthebestsolutionhasevenbeenattempted.Agoodapproach is to allow the optimization to continuefor a "long enough" time and check to see if thepopulation of solutions converge. After that, theuser could try the optimization procedure severaltimes to make sure that the answers agree (or areclose) and that the first answer is not a false orsub-optimalone.

5.1StepsforusingoptimizationinExtend:

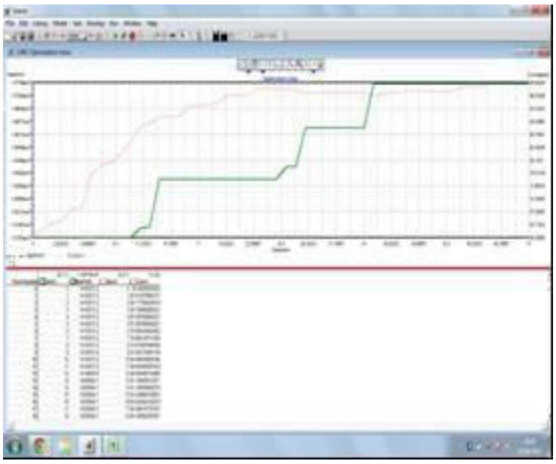
"problem"isusuallystatedasanObjective
1.Openamodelweneedtooptimize.
function;equivalenttoacostorprofitequation·
2. Openthe Genericlibrarybygoingtothe
whichthemodeleristryingtominimizeorLibrarymenu,

maximizewithoutgoingthroughthetediousjob.
3.PlaceanEvolutionaryOptimizerblockonthe
model.4.Defineacost or profitequation(alsocalledthe objective function) that you would like tooptimize.

4. Determinewhichparametersyouneedforthatequation.
5. Drag the variables that you need onto theclosedevolutionaryOptimizer blockonthemodel.
6. SetthelimitsforthosevariablesintheOptimizer'sVariablestable.
7. PuttheprofitorcostequationintheOptimizer'sdialog.
8. Ifavariableneedstobeconstrainedbyothervariables,addconstraintequations.
9. Set the optimizer defaults for random or non-random model.
10. ClickontheResultstabandclickthe"RunNow"button.

RESULTOFTHEEVOLUTIONARYOPTIMISATIONOFTHE MODEL:

Thesimulationmodelwasoptimisedusing the Evolutionary optimisation and theresultsareshowninthe belowTable.1



CONCLUSION:

The optimized inventory levels required for theproduct model 1 is 160 components and for theproduct model 2 is 195 components for aconstant customer demand resulted from theEvolutionary optimization to achieve themaximumprofit.

FURTHERWORK:

Based on the results we need to develop analgorithm which can do the evolutionaryoptimizationandthealgorithmtobe validatedusingthemodel.

REFERENCES:

1) **YoonchangandHarrismakatsoris**,Supplychain modelingusingsimulation

2) **J. Sudhir Ryan Daniel and ChandrasekharanRajendran**(2009),Asimulation-basedgeneticalgorithmforinventoryoptimizationinaserialsupplychain

3) **DavidNaso,MicheleSurico,andBiagioTurchiano ,UzayKaymak**,Genetic algorithms forsupply-chainscheduling:acasestudyinthedistribution ofready-mixedconcrete

4) **Takayuki yoshizumi ,Hiroyuki okano**(2007),Asimulation-basedalgorithmforsupplychainoptimization

5) **SandipanKarmakar and BiswajitMahanty**(2009), Minimizing Make span for a Flexible FlowShop SchedulingProbleminaPaintCompany

6) **L. Wang and D.-Z. Zheng**(2003), An EffectiveHybridHeuristicforFlowShopScheduling
7) Supplychainmanagement -EBook
8) OptimizationMethods:IntroductionandBasic