Problema de Balanceamento e Designação de Trabalhadores em Linha de Produção: uma abordagem via metaheurística Clustering Search

Antonio A. Chaves[1], Cristobal Miralles[2], Luiz A.N. Lorena[1]

[1] LAC – Instituto Nacional de Pesquisas Espaciais (INPE)
Av. dos Astronautas, 1758 São José dos Campos – SP – Brasil

[2] Dpto. Organización de Empresas. Universidad Politécnica de Valencia
Camí Vera s/n, 46022, Valencia – Espanha

{chaves,lorena}@lac.inpe.br, cmirales@omp.upv.es

Resumo. *O Problema de Balanceamento e Designação de Trabalhadores em Linha de Produção (ALWABP) consiste em otimizar a alocação de trabalhadores e tarefas às estações de trabalho em uma linha de produção com pessoas deficientes, objetivando aumentar a produtividade de tais linhas e tornar as deficiências individuais tão transparente quanto possível. Este trabalho propõe uma nova abordagem para resolver o ALWABP, utilizando a metaheurística híbrida* Clustering Search *(CS). A validação das soluções obtidas será através da comparação com os resultados encontrados pelo CPLEX.*

Abstract. *The Assembly Line Worker Assignment and Balancing Problem (ALWABP) consists of optimizing the allocation of workers and tasks to stations in an assembly line with disabled people. The goal is to increase the productivity of the assembly line but always respecting the worker limitations; trying to make transparent the individual disabilities. This paper proposes a new approach to solve the ALWABP using the hybrid metaheuristic Clustering Search (CS). The validation of the obtained solutions will be through the comparison with the results found by the CPLEX.*

1. Introdução

A Organização Mundial de Saúde (OMS) estima que 10% da população mundial, cerca de 610 milhões de pessoas, são portadores de algum tipo de deficiência. Destas, mais da metade estão na idade ativa de trabalho e sofrem com altas taxas de desemprego (variando de 13% no Reino Unido a 80% em países em desenvolvimento).

Práticas atuais para o tratamento de indivíduos com deficiências físicas e mentais prescrevem atividades de trabalho, objetivando que estes indivíduos tenham uma vida mais longa e também uma maior integração destes cidadãos à sociedade (Belamy et al., 1979). Como forma de facilitar a inclusão destes trabalhadores no mercado de trabalho, alguns países adotaram a estratégia de criar Centros de Trabalho para Deficientes (CTD's). Estes centros funcionam como uma primeira etapa na integração destas pessoas que, eventualmente, serão absorvidas pelo mercado normal de trabalho.

Em Miralles et al. (2007) é mostrado como a utilização de linhas de produção nos CTD's provê muitas vantagens, sendo que, a divisão tradicional do trabalho em tarefas únicas pode se tornar uma ferramenta perfeita para fazer invisíveis determinadas deficiências do trabalhador. Além disso, uma atribuição apropriada da tarefa também pode transformar-se em um bom método terapêutico para reabilitação das deficiências. Mas algumas restrições específicas relacionadas à variabilidade dos tempos surgem neste ambiente, e então o procedimento de balanceamento aplicado neste ambiente deveria conciliar os seguintes objetivos (que não devem ser vistos como contraditórios, mas complementares): (1) maximizar a eficiência da linha de produção balanceando a carga de trabalho atribuída a cada trabalhador disponível em cada estação; (2) satisfazer e respeitar as restrições existentes neste ambiente devido aos fatores humanos ao atribuir tarefas aos trabalhadores.

Após analisar alguns CTD's, Miralles et al. (2007) definiram um novo problema de linha de produção, chamado Problema de Balanceamento e Designação de Trabalhadores em Linhas de Produção (ALWABP, do inglês: Assembly Line Worker Assignment and Balancing Problem). Uma das principais características definidas neste trabalho é que existem diferentes tempos de execução por pares (trabalhador, tarefa), portanto, é necessário definir uma matriz de tempo para as tarefas ao invés de uma lista. Nesta matriz algumas incompatibilidades podem aparecer dependendo do tipo de deficiência do trabalhador.

O ALWABP pode ser classificado como um problema NP-difícil, pois este é uma generalização do Problema Simples de Balanceamento da Linha de Produção (SALB, do inglês: Simple Assembly Line Balancing Problem), onde cada tarefa possui um tempo de execução fixo independente do trabalhador que a executa. Sendo o SALBP classificado como NP-difícil (Scholl e Becker, 2006). Esta característica justifica a aplicação de metaheurísticas ao ALWABP visando conseguir bons resultados em um tempo computacional competitivo.

Assim como ocorre no SALBP, quando deseja-se minimizar o número de estações, o problema é chamado ALWABP-1 e quando o objetivo é minimizar o tempo de ciclo, o problema é chamado ALWABP-2, esta última situação é mais comum nos CTD's. Por esta razão, Miralles et al. (2007) apresentam um modelo matemático para o ALWABP-2 e um estudo de caso em um CTD espanhol.

Neste trabalho propõe-se utilizar a metaheurística híbrida Clustering Search (CS) para resolver de forma aproximada o ALWABP-2. O CS consiste em detectar dinamicamente regiões promissoras de busca baseando-se na freqüência em que são amostradas nestas regiões as soluções geradas por uma metaheurística. Estas regiões promissoras devem ser exploradas tão logo sejam descobertas, através de métodos de busca local. O software CPLEX é utilizado para resolver as formulações matemáticas do ALWABP-2, objetivando validar os resultados computacionais obtidos pelo CS.

O restante do trabalho está organizado da seguinte forma. A seção 2 apresenta o CS aplicado ao ALWABP. A seção 3 apresenta os resultados computacionais e na seção 4 são descritas algumas conclusões a respeito desse trabalho.

2. Clustering Search aplicado ao ALWABP-2

A metaheurística híbrida Clustering Search (CS) é baseada no Evolutionary Clustering Search (ECS), proposto por Oliveira e Lorena (2007). O CS consiste em dividir o espaço de busca em regiões, ou seja, sub-espaços de busca definidos por uma relação de vizinhança, e por meio de um processo de agrupamento de soluções detectar regiões supostamente promissoras.

O CS procura localizar regiões promissoras através do enquadramento destas por *clusters*. Um *cluster* é formado pelo **centro** c_i e pelo **volume** v_i. O centro c_i é uma solução que representa o *cluster i*, identificando a sua localização dentro do espaço de busca. Ao invés de armazenar todas as soluções agrupadas no *cluster*, apenas parte das informações destas soluções são atribuídas ao centro do *cluster*. O volume v_i é a quantidade de soluções geradas pela metaheurística e agrupadas no *cluster i*. Este volume é o que determinará quando um *cluster* se torna promissor.

O CS consiste em duas etapas que vão sendo executadas iterativamente. Na primeira etapa gera-se uma solução por meio de uma metaheurística. Na segunda etapa realiza-se o processo de agrupamento, onde a solução gerada é agrupada no *cluster* mais similar e caso este *cluster* seja considerado promissor, aplica-se uma heurística de busca local neste.

A metaheurística responsável pela geração de soluções para o processo de agrupamento será o *Simulated Annealing* (SA) (Kirkpatrick, 1983), a qual é capaz de gerar um grande número de soluções diferentes para este processo.

O SA parte de uma solução inicial aleatória, a qual é obtida escolhendo aleatoriamente as alocações das tarefas para as estações e de trabalhadores disponíveis para estações. Os passos seguintes seguem o esquema do algoritmo tradicional do SA. Dada uma temperatura T, o algoritmo aleatoriamente seleciona um dos movimentos para uma vizinhança e computa a variação da função objetivo. Se este melhorar a solução corrente o movimento é aceito, caso contrário, existe uma probabilidade de aceitação do movimento que é menor em temperaturas baixas.

Três movimentos diferentes podem ser definidos para compor tipos distintos de vizinhança, chamados N^1, N^2 e N^3, de uma solução. N^1 é obtido pela troca de alocação de dois trabalhadores de diferentes estações. N^2 pela troca de alocação de duas tarefas de diferentes estações. N^3 pela retirada de uma tarefa de uma estação e adição em outra estação.

Os parâmetros de controle do SA são a razão de resfriamento α, o número de iterações para cada temperatura (SA_{max}) e a temperatura inicial T_0. Neste trabalho foi utilizado $\alpha = 0{,}95$; $SA_{max} = 1000$; e $T_0 = 1000000$.

As soluções geradas pelo SA são passadas para o processo de agrupamento. O *cluster* que possuir a menor distância em relação à solução gerada pelo SA é considerado o *cluster* mais similar. A informação da solução gerada pelo SA deve causar uma perturbação (assimilação) no centro do *cluster* mais similar. A medida de distância utilizada neste trabalho foi o número de tarefas atribuídas para estações diferentes entre a solução SA e o centro do *cluster*, sendo que, quanto maior o número

de tarefas atribuídas a diferentes estações entre as duas soluções, maior será a distância entre elas e menor será a similaridade.

No processo de assimilação será utilizado o método *path-relinking* (Glover, 1996), o qual realiza movimentos exploratórios na trajetória que interconecta o centro do *cluster* mais similar (c_i) e a solução gerada pelo SA (s_k). Assim sendo, o próprio processo de assimilação já é uma forma de busca local dentro do *cluster*, pois o centro vai ser deslocado para a melhor solução avaliada nessa trajetória.

Após agrupar a solução ao cluster mais similar e aplicar o processo de assimilação no centro deste, é necessário verificar se o *cluster* já pode ser considerado promissor. Um *cluster* se torna promissor quando o volume atinge um certo limitante λ. Caso o volume do *cluster* seja maior que esse limitante significa que um certo padrão de informação se tornou predominante no processo de busca, sendo o centro deste *cluster* explorado através de uma heurística específica para o problema.

Neste trabalho foram implementadas duas heurísticas de busca local para o ALWABP-2, objetivando melhorar a solução presente no centro de um cluster promissor. A primeira heurística, chamada SWAP, consiste em examinar todos as possíveis trocas de duas tarefas que estão atribuídas a diferentes estações. A melhor troca é realizada e o processo é repetido novamente até que não ocorra mais melhora. A segunda heurística, chamada SHIFT, consiste em examinar todas as relações de tarefas e estações. A heurística remove uma tarefa de uma estação e insere esta em uma outra estação. O melhor movimento é realizado e todo o processo é repetido novamente até não ocorrer mais melhora. Caso as heurísticas encontrem uma solução que seja melhor que o centro do cluster, este é atualizado com a solução encontrada.

3. Resultados Computacionais

O CS para ALWABP foi codificado em C++ e os testes computacionais executados em um Pentium IV 3 GHz com 512 MB de memória. Os experimentos foram realizados com objetivo de validar a abordagem proposta, mostrando que o CS pode ser competitivo para resolução do ALWABP-2.

Para avaliar a qualidade das soluções encontradas pelo CS, executou-se o CPLEX versão 10.0.1 para cada uma das 320 instâncias. Devido ao fato do ALWABP ser um problema NP-difícil, há uma grande dificuldade para encontrar a solução ótima, especialmente para o conjunto de problemas maiores. Na maioria das instâncias de Wee-Mag e Tonge, o CPLEX terminou a execução devido ao estouro de memória do computador, sendo que, as soluções inteiras encontradas estavam com *gaps* acima de 90% entre o limitante inferior e superior.

A Tabela 1 apresenta os resultados para os problemas Roszieg e Heskia. Para as 160 instâncias presentes neste dois conjuntos de problemas, o CS encontrou a solução ótima em todos os casos, mostrando que este é um método interessante para resolução do ALWABP. O CPLEX consegue resolver todas as instâncias de Roszieg e Heskia, porém, o CPLEX é sensível em relação à variação do número de trabalhadores, uma vez que, o tamanho da matriz tarefa/trabalhador é aumentado e conseqüentemente a complexidade também. No CS esse efeito praticamente não é observado, sendo que, a variação de tempo é pequena em relação ao aumento do número de trabalhadores. O SA

sem processo de agrupamento também obteve bons resultados, porém não encontrou todas as soluções ótimas e as soluções médias são piores se comparadas com as do CS.

Tabela 1 – Resultados Computacionais (problemas Roszieg e Heskia)

	Tam	*NrT*	*Var*	*Inc*	CPLEX			CS			SA		
					SI	*Tempo*	*gap*	*Sol**	*Sol*	*Tempo*	*Sol**	*Sol*	*Tempo*
Roszieg	25	4	L1	I10	20,10	4,72	0	20,10	20,19	5,51	20,10	20,32	4,73
				I20	31,50	3,77	0	31,50	33,31	5,42	31,70	40,06	4,71
			H3	I10	28,10	5,46	0	28,10	28,14	5,49	28,20	28,19	4,68
				I20	28,00	4,27	0	28,00	28,12	5,49	28,00	28,25	4,69
		6	L1	I10	9,70	332,21	0	9,70	10,16	6,21	9,80	10,26	5,02
				I20	11,00	281,03	0	11,00	11,89	6,18	11,00	12,15	5,02
			H3	I10	16,00	390,07	0	16,00	16,28	6,22	16,10	16,40	5,01
				I20	15,10	686,75	0	15,10	15,36	6,23	15,30	15,58	5,03
Heskia	28	4	L1	I10	102,30	6,56	0	102,30	103,34	6,20	103,10	104,69	4,76
				I20	122,60	5,78	0	122,60	123,50	6,15	122,80	126,17	4,76
			H3	I10	172,50	6,70	0	172,50	173,23	6,18	172,50	173,91	4,74
				I20	171,20	7,30	0	171,20	171,71	6,19	171,20	171,91	4,75
		7	L1	I10	34,90	177,36	0	34,90	35,95	7,58	35,00	36,64	5,57
				I20	42,60	216,38	0	42,60	44,28	7,58	43,40	44,58	5,55
			H3	I10	75,20	260,93	0	75,20	76,17	7,56	75,70	76,60	5,51
				I20	67,20	341,83	0	67,20	70,09	7,58	67,20	70,38	5,53

A Tabela 2 apresenta os resultados para os problemas Wee-Mag e Tonge. Os resultados obtidos pelo CPLEX para estes problemas não foram satisfatórios, sendo que, para as 160 instâncias testadas foi possível encontrar apenas soluções viáveis com gaps muito grandes, e em alguns casos, as soluções obtidas possuem atribuições incompatíveis entre tarefa e trabalhador. Além disso, em algumas instâncias o CPLEX não consegue encontrar nenhuma solução viável antes que ocorra estouro de memória do computador. Os resultados obtidos pelo CS foram melhores que as soluções viáveis providas pelo CPLEX em todos os casos. O tempo computacional do CS foi de aproximadamente 100 segundos nos piores casos, enquanto que no CPLEX o tempo computacional é muito alto e, além disso, o CPLEX é interrompido devido ao estouro de memória do computador na maioria das instâncias. Novamente o SA sem o processo de agrupamento obteve resultados piores que o CS.

Tabela 2 – Resultados Computacionais (problemas Tonge e Wee-Mag)

	Tam	*NrT*	*Var*	*Inc*	CPLEX				CS			SA		
					SI	*SNI*	*Tempo*	*gap*	*Sol**	*Sol*	*Tempo*	*sol**	*sol*	*tempo*
Tonge	70	10	L1	I10	144,6	10,56	20222,9	92,6	104,4	168,3	65,12	285,0	854,9	19,93
				I20	162,4	12,03	23863,7	92,4	126,7	216,2	64,03	134,1	851,1	20,04
			H3	I10	276,7	16,45	18286,1	93,9	178,7	273,6	64,27	188,4	870,7	20,41
				I20	292,8	18,59	21241,8	93,6	185,0	295,3	64,32	272,1	881,7	20,38
		17	L1	I10	107,2	1,61	57480,3	98,5	54,4	95,6	96,55	343,1	940,5	24,73
				I20	772,2	1,74	44656,3	98,7	69,2	133,4	96,34	533,0	982,8	24,72
			H3	I10	231,4	2,63	54316,3	98,9	104,4	184,4	96,22	467,1	978,3	25,14
				I20	376,8	2,64	57772,4	98,9	114,7	183,5	96,13	668,2	999,5	25,08
Wee-Mag	75	11	L1	I10	42,4	2,80	17884,8	93,3	31,3	36,9	76,73	31,9	62,4	21,01
				I20	50,8	3,40	21507,3	93,2	37,3	42,6	76,26	38,0	49,1	21,31
			H3	I10	77,4	4,92	17227,8	93,6	55,2	63,8	76,51	55,9	89,4	21,35
				I20	72,9	4,69	17039,1	93,5	51,8	62,4	76,46	53,6	69,3	21,35
		19	L1	I10	373,8	0,84	33323,6	99,1	16,8	21,5	111,38	17,4	544,2	26,52
				I20	1258,5	0,85	43084,5	99,8	19,3	24,3	111,63	19,7	494,0	26,92
			H3	I10	659,5	0,90	51772,6	99,4	27,7	36,2	111,87	29,5	458,2	27,09
				I20	470,0	0,87	56884,9	99,4	27,0	35,4	111,60	28,1	494,8	27,06

4. Considerações Finais

Este artigo apresentou uma solução para o Problema de Balanceamento e Designação de Trabalhadores em Linha de Produção (ALWABP, do inglês Assembly Line Worker Assignment and Balancing Problem) utilizando a metaheurística híbrida Clustering Search (CS). Os resultados mostram que o CS é muito competitivo para resolução deste problema em tempos computacionais razoáveis. Uma vantagem adicional desta abordagem é o fato de não utilizar nenhum solver comercial, o que facilitará uma implementação real para ser usada pelos seus beneficiários finais: os CTD's.

No senso prático, os tempos computacionais baixos encontrados pelo CS possibilitam um rápido balanceamento das linhas de produção. Isto é muito importante se levarmos em conta que, devido ao alto absentismo e as consultas físicas e psicológicas periódicas, o responsável pelo CTD conhece somente no início de cada dia de trabalho quais trabalhadores estão disponíveis. Portanto abordagens como o CS, que provêem boas soluções em tempos computacionais pequenos, são muito desejáveis para montar uma linha de produção diariamente. Para as rotinas de rotação de trabalhadores, que são necessárias no ambiente dos CTD's, também é importante ter um ágil balanceamento da linha para que se tenha uma vasta combinação de soluções facilmente.

Os trabalhos futuros que podem ser feitos sobre o ALWABP englobam desde uma implementação real deste problema, até a análise de outras metaheurísticas para gerar soluções para o processo de agrupamento do CS, e a implementação de novos métodos de busca local.

Referências

Bellamy, G. T., Horner, R. H. e INMAN, D. P., Vocational Habilitation of Severely Retarded Adults: A direct Service Technology. University Press, Baltimore-USA, 1979.

Glover, F., Tabu search and adaptive memory programming: Advances, applications and challenges, em Barr, R. S., Helgason, R. V. e Kennington, J. L. (Eds.), Interfaces in Computer Science and Operations Research. Kluwer, 1–75, 1996.

Hoffmann, T. R., Assembly line balancing: A set of challenging problems. International Journal of Production Research, 28, 1807–1815, 1990.

Kirkpatrick, S., Gellat, D. C. e Vecchi, M. P., Optimization by simulated annealing. Science, v. 220, 671–680, 1983.

Miralles, C., García-Sabater, J., Andrés, C. e Cardós, M., Advantages of assembly lines in sheltered work centres for disabled. a case study. International Journal of Production Economics, 187-197, 2007.

Oliveira, A. C. M. e Lorena, L. A. N., Hybrid evolutionary algorithms and clustering search, em Grosan, C., Abraham, A. e Ishibuchi, H. (Eds.), Hybrid Evolutionary Systems - Studies in Computational Intelligence. Springer SCI Series, 81–102, 2007.

Scholl, A. e Becker, C., State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. European Journal of Operational Research, 168, 666–693, 2006.