

Um modelo de programação linear inteira para organização de fóruns empresariais

José Francisco Moreira Pessanha
professorjfm@hotmai.com
Uerj

Narcisa Maria Gonçalves dos Santos
narcisa.santos@gmail.com
Uerj

Resumo: Associações empresariais costumam promover fóruns entre empresários e empreendedores de diferentes segmentos. Uma estratégia para estabelecer contatos entre os participantes consiste em organizá-los em pequenos grupos e alocá-los em mesas, de tal forma que os participantes em um mesmo grupo possam divulgar suas empresas, apresentar projetos e trocar ideias e experiências. Após um determinado intervalo de tempo, uma nova sessão é iniciada e novos grupos de participantes são formados e alocados em outras mesas para que novos contatos sejam realizados. A repetição desta dinâmica até o encerramento do fórum permite que cada participante realize um grande número de contatos. Contudo, visando ampliar as redes de relacionamentos é fundamental evitar os reencontros de participantes ao longo das sessões do evento. Neste trabalho apresenta-se um modelo de Programação Linear Inteira capaz de organizar um roteiro com a sequência de mesas a serem visitadas por cada participante com o objetivo de evitar ou minimizar os reencontros. O modelo foi implementado em ambiente R e conta com uma interface em planilha eletrônica para entrada de dados e apresentação de resultados. Os resultados gerados pelo modelo são ilustrados por meio de um experimento computacional.

Palavras Chave: Programação Inteira - Fóruns empresariais - Programa R - Planilha eletrônica -

1. INTRODUÇÃO

Uma rede de contatos é um poderoso instrumento para alavancar o crescimento de qualquer negócio. A conectividade favorece o surgimento de ideias (Johnson, 2011) e iniciativas com forte potencial para promover a inovação e o crescimento econômico, beneficiando toda a sociedade. Por esta razão, não raro, as associações empresariais costumam promover fóruns que estimulem a conversação e a colaboração entre empresários e empreendedores de diferentes segmentos.

Uma estratégia para estabelecer contatos entre os numerosos participantes de um fórum de negócios consiste em organizá-los em pequenos grupos e aloca-los em mesas, de tal forma que os participantes em um mesmo grupo (mesa) possam divulgar suas empresas, apresentar projetos e trocar ideias e experiências com os demais participantes do grupo. Após um determinado intervalo de tempo, suficiente para que os membros dos grupos se conheçam, uma nova sessão é iniciada e novos grupos de participantes são formados e alocados em outras mesas para que novos contatos sejam realizados. A repetição desta dinâmica até o encerramento do fórum permite que cada participante realize um grande número de contatos aleatórios. Contudo para que esta estratégia seja exitosa em ampliar as redes de relacionamentos é fundamental evitar os reencontros de participantes na formação dos novos grupos ao longo das sessões do evento. Para evitar ou minimizar os reencontros é necessário organizar previamente um roteiro com a sequência de mesas a serem visitadas por cada participante.

A determinação de roteiros sem reencontros dos participantes guarda alguma semelhança com o problema do caixeiro viajante, um problema clássico da Programação Linear Inteira - PLI (Ragsdale, 2004) em que o objetivo consiste em estabelecer uma sequência de localidades a serem visitadas, de tal forma que cada localidade seja visitada apenas uma vez. Trata-se de um problema de difícil solução e que requer grande esforço computacional para ser resolvido. A PLI tem vasta aplicação prática e fornece desde soluções aos problemas de roteirização e localização de instalações (Ragsdale, 2004) até soluções para o quebra cabeça Sudoku (Bartlett et al, 2008) e a organização dos convidados à uma festa de casamento com base nas afinidades entre eles (Bellows & Peterson, 2012).

No presente texto descreve-se um modelo de Programação Linear Inteira concebido para estabelecer os roteiros das mesas a serem visitadas pelos participantes de um fórum de negócios, de tal forma que o número de reencontros ao longo do evento seja minimizado. O modelo foi implementado em linguagem R (R Core Team, 2014) e conta com uma interface para entrada de dados e visualização de resultados programada em planilha MS Excel (Ragsdale, 2004). O artigo está organizado em cinco seções. A formulação do Problema de Programação Linear Inteira encontra-se descrita na seção 2 por meio de um exemplo simples envolvendo apenas seis participantes e três mesas. Na sequência, na seção 3, tem-se uma breve apresentação da aplicação dos recursos do R e do MS Excel na implementação computacional do modelo proposto. Na seção 4 apresentam-se alguns resultados selecionados oriundos da aplicação do modelo proposto na organização de um evento real com cerca de 100 empresários. Por fim, na seção 5 apresentam-se as principais conclusões do trabalho.

2. FORMULAÇÃO DO PROBLEMA

Considere um evento com n participantes, organizados em m mesas e p pessoas por mesa. Uma condição necessária para que não haja reencontros é que o número de pessoas por mesa seja menor que o número de mesas ($p < m$). Um participante $i \forall i=1, n$ pode ocupar uma mesa $j \forall j=1, m$ e para representar esta possibilidade considera-se uma variável binária x_{ij} que assume valor igual a 1 se o participante i ocupa a mesa j , caso contrário x_{ij} é igual a 0. Para um

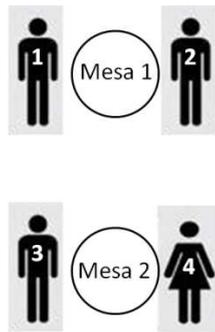
evento com n participantes e m mesas há um total de nm variáveis binárias $x_{ij} \in \{0,1\} \forall i=1,n$ e $j=1,m$. O número de participantes em uma mesa j deve ser igual a p , logo os valores de $x_{ij} \forall i=1,n$ devem satisfazer a seguinte condição em cada mesa j :

$$\sum_{i=1}^n x_{ij} = p \forall j=1,m \quad (1)$$

Adicionalmente, cada participante i só pode ocupar uma mesa em cada sessão, logo os valores de $x_{ij} \forall j=1,m$ devem satisfazer a seguinte condição em cada participante:

$$\sum_{j=1}^m x_{ij} = 1 \forall i=1,n \quad (2)$$

As mesas são numeradas de 1 a m e ao chegarem no evento os participantes escolhem as mesas aleatoriamente e definem uma configuração inicial. Dado que cada mesa comporta p pessoas, os participantes que ocupam a mesa 1 são identificados pelos números inteiros de 1 até p , os participantes da mesa 2 recebem os números $(p+1)$ até $(p+p)$ e assim sucessivamente até o n -ésimo participante na mesa m . Na Figura 1a tem-se uma ilustração da configuração inicial da alocação dos participantes nas mesas e ao lado, na Figura 1b, a representação matricial desta configuração.



$$\begin{aligned} x_{11}=1, x_{21}=1, x_{31}=0, x_{41}=0, x_{51}=0, x_{61}=0 \\ x_{12}=0, x_{22}=0, x_{32}=1, x_{42}=1, x_{52}=0, x_{62}=0 \\ x_{13}=0, x_{23}=0, x_{33}=0, x_{43}=0, x_{53}=1, x_{63}=1 \end{aligned}$$

a) $n=6$ participantes, $m=3$ mesas, $p=2$ lugares

b) Matriz de alocação

Figura 1: Configuração inicial da ocupação das mesas.

Então, inicia-se a primeira sessão de contatos e após 20 minutos os participantes devem trocar de mesas, de tal forma que na nova configuração os reencontros sejam minimizados. Assim, a nova configuração de n participantes em m mesas é determinada pelo seguinte Problema de Programação Linear Inteira - PPLI, cujo objetivo consiste em minimizar o número de reencontros:

$$\text{Min}_{x,y} \sum_{k \in \{1, \dots, m(m-1)\}} y_k \quad (3)$$

s.a.

$$\sum_{i=1}^n x_{ij} = p \forall j=1,m \quad (4)$$

$$\sum_{j=1}^m x_{ij} = 1 \forall i=1,n \quad (5)$$

$$\sum_{i \in \text{mesa } k} x_{ij} \leq 1 + y_{l+(m-1)(k-1)} \forall k=1,m, \forall j=1,m \text{ e } j \neq k, \forall l=1,(m-1) \quad (6)$$

$$x_{ij} \in \{0,1\} \forall i=1,n, \forall j=1,m \quad (7)$$

$$y_j \in \{0,1,2,3,\dots\} \forall j=1,m(m-1) \quad (8)$$

As m restrições na equação (4) significam que cada mesa deve ter p participantes. As n restrições em (5) indicam que cada participante pode ocupar apenas uma mesa em cada sessão. As $m(m-1)$ restrições em (6) impedem os reencontros de participantes que ocuparam a mesma mesa na configuração inicial. A restrição em (7) indica que as variáveis x_{ij} são binárias e a restrição em (8) informa que as variáveis y_k são inteiras. As variáveis binárias indicam a presença ou ausência do participante i na mesa j . Por sua vez, a variável y_k conta o número de reencontros e, portanto, a soma destas variáveis deve ser minimizada, conforme indicado na função objetivo em (3). No caso com $n=6$ participantes, $m=3$ mesas e $p=2$, a nova configuração é determinada pelo seguinte PPLI:

$$\text{Min } y_1 + y_2 + y_3 + y_4 + y_5 + y_6$$

(9)

s.a.

$$x_{11} + x_{21} + x_{31} + x_{41} + x_{51} + x_{61} = 2 \quad (\text{mesa 1}) \quad (10)$$

$$x_{12} + x_{22} + x_{32} + x_{42} + x_{52} + x_{62} = 2 \quad (\text{mesa 2}) \quad (11)$$

$$x_{13} + x_{23} + x_{33} + x_{43} + x_{53} + x_{63} = 2 \quad (\text{mesa 3}) \quad (12)$$

$$x_{11} + x_{12} + x_{13} = 1 \quad (\text{participante 1}) \quad (13)$$

$$x_{21} + x_{22} + x_{23} = 1 \quad (\text{participante 2}) \quad (14)$$

$$x_{31} + x_{32} + x_{33} = 1 \quad (\text{participante 3}) \quad (15)$$

$$x_{41} + x_{42} + x_{43} = 1 \quad (\text{participante 4}) \quad (16)$$

$$x_{51} + x_{52} + x_{53} = 1 \quad (\text{participante 5}) \quad (17)$$

$$x_{61} + x_{62} + x_{63} = 1 \quad (\text{participante 6}) \quad (18)$$

$$x_{12} + x_{22} \leq y_1 + 1 \quad (\text{evita reencontro entre participantes 1 e 2 na mesa 2}) \quad (19)$$

$$x_{13} + x_{23} \leq y_2 + 1 \quad (\text{evita reencontro entre participantes 1 e 2 na mesa 3}) \quad (20)$$

$$x_{31} + x_{41} \leq y_3 + 1 \quad (\text{evita reencontro entre participantes 3 e 4 na mesa 1}) \quad (21)$$

$$x_{33} + x_{43} \leq y_4 + 1 \quad (\text{evita reencontro entre participantes 3 e 4 na mesa 3}) \quad (22)$$

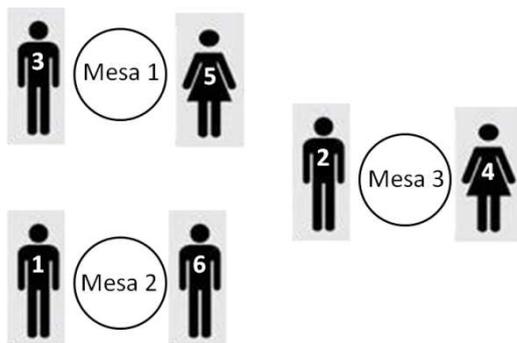
$$x_{51} + x_{61} \leq y_5 + 1 \quad (\text{evita reencontro entre participantes 5 e 6 na mesa 1}) \quad (23)$$

$$x_{52} + x_{62} \leq y_6 + 1 \quad (\text{evita reencontro entre participantes 5 e 6 na mesa 2}) \quad (24)$$

$$x_{11} = 0, x_{21} = 0, x_{32} = 0, x_{42} = 0, x_{53} = 0, x_{63} = 0 \quad (\text{impede revisitar mesas}) \quad (25)$$

$$x_{ij} \in \{0,1\} \quad \forall i=1,6, \quad \forall j=1,3 \quad y_j \in \{0,1,2,3\} \quad \forall j=1,6 \quad (26)$$

Vale ressaltar que o conjunto formado pelas restrições (19) até (24) foi gerado diretamente da solução ilustrada na Figura 1.b e tem por objetivo evitar os reencontros entre participantes na configuração inicial. A solução do PPLI acima resulta na solução ótima ilustrada na Figura 2. Na nova solução a soma $y_1 + \dots + y_6$ é nula, logo, não houve reencontros.



$$\begin{aligned} x_{11}=0, x_{21}=0, x_{31}=1, x_{41}=0, x_{51}=1, x_{61}=0 \\ x_{12}=1, x_{22}=0, x_{32}=0, x_{42}=0, x_{52}=0, x_{62}=1 \\ x_{13}=0, x_{23}=1, x_{33}=0, x_{43}=1, x_{53}=0, x_{63}=0 \\ y_1=0, y_2=0, y_3=0, y_4=0, y_5=0, y_6=0 \end{aligned}$$

a) $n=6$ participantes, $m=3$ mesas, $p=2$ lugares

b) Matriz de alocação

Figura 2: Configuração na segunda sessão.

Após 20 minutos mais uma troca de mesas deve ser realizada e uma nova configuração deve ser encontrada. Novamente o PPLI deve ser resolvido, porém com um conjunto de restrições adicionais que evitem os reencontros com todas as configurações anteriores. Tal conjunto de restrições é semelhante ao conjunto em (6) contendo $m(m-1)$ restrições, porém é derivado diretamente da configuração da sessão imediatamente anterior. Assim, ao longo da evolução do evento o conjunto de restrições do PPLI é acrescido de $m(m-1)$ restrições a cada sessão. Adicionalmente, todas as variáveis x_{ij} que assumiram valores unitários nas sessões anteriores são fixadas em zero nas novas sessões para que cada participante visite cada mesa uma única vez. Para obter a configuração da terceira e última sessão do caso com $n=6$ participantes, $m=3$ mesas e $p=2$ basta observar a configuração ilustrada na Figura 2 para gerar o seguinte conjunto de restrições a serem adicionadas ao PPLI definido a partir da função objetivo em (9):

$$x_{32} + x_{52} \leq y_7 + 1 \text{ (evita reencontro entre participantes 3 e 5 na mesa 2)} \quad (27)$$

$$x_{33} + x_{53} \leq y_8 + 1 \text{ (evita reencontro entre participantes 3 e 5 na mesa 3)} \quad (28)$$

$$x_{11} + x_{61} \leq y_9 + 1 \text{ (evita reencontro entre participantes 1 e 6 na mesa 1)} \quad (29)$$

$$x_{13} + x_{63} \leq y_{10} + 1 \text{ (evita reencontro entre participantes 1 e 6 na mesa 3)} \quad (30)$$

$$x_{21} + x_{41} \leq y_{11} + 1 \text{ (evita reencontro entre participantes 2 e 4 na mesa 1)} \quad (31)$$

$$x_{22} + x_{42} \leq y_{12} + 1 \text{ (evita reencontro entre participantes 2 e 4 na mesa 2)} \quad (32)$$

3. IMPLEMENTAÇÃO COMPUTACIONAL

Por meio do R foi possível construir um programa capaz de automatizar a montagem do PPLI em cada sessão do evento e resolve-lo com o pacote Rsymphony (Harter et al, 2016), uma interface de alto nível entre o R e o COIN-OR *Symphony solver*.

Todo o código em R encontra-se em um único arquivo texto com extensão Rexec. Algumas manipulações no *path* do sistema permitem associar a extensão Rexec ao interpretador do R (Rscript.exe) e assim o arquivo Rexec, com o programa em R, funciona como um arquivo executável que pode ser acionado por uma macro do MS Excel.

Para associar a extensão Rexec ao Rscript procure pela linha de comando (CMD), pressione o respectivo ícone com o botão direito do *mouse* e escolha a opção *Executar como administrador*, conforme ilustrado na Figura 3. Na sequência, abre-se a janela da linha de comando ilustrada na Figura 4, na qual devem ser inseridas as seguintes linhas de comando a partir do *prompt*:

```
ASSOC .Rexec=RscriptExecutable
```

```
FTYPE RscriptExecutable=C:\ProgramFiles\RR-3.3.1\bin\RScript.exe %1 %*
```

O *path* C:\ProgramFiles\RR-3.3.1\bin\RScript.exe indicado acima deve corresponder ao local onde está instalado o arquivo RScript.exe sem as barras “\”.

Ao final, acesse as variáveis de ambiente do sistema por meio do *Painel de controle e Definições Avançadas* e insira a extensão “;Rexec” no final do *path*. Mais detalhes sobre a execução de arquivo Rexec podem ser encontrados no *post Making R Files Executable (under Windows)* disponível no R-bloggers (Meissner, 2015).

Os dados de entrada/saída do programa são armazenados em arquivos textos. Os dados de entrada incluem o nº de participantes do evento, o nº de pessoas por mesa e o nº de sessões. Já os resultados são gravados em arquivos .XLSX e incluem a sequência de mesas visitadas por cada participante (roteiros), um quadro com as alocações dos participantes em cada sessão e o nº de reencontros.

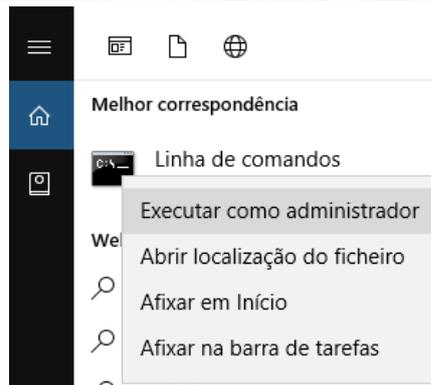


Figura 3: Execução da linha de comando como administrador.

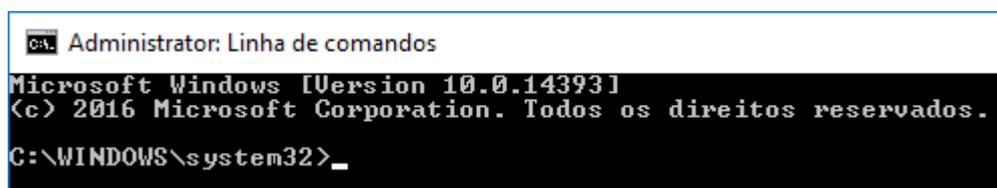


Figura 4: Linha de comando.

Para facilitar a utilização do programa foi construída uma planilha MS Excel com botões e macros (Jellen & Syrstad, 2008) programados para preparar o arquivo de entrada de dados, disparar a execução do arquivo Rexec e importar os resultados gerados pelo R para dentro da planilha. A comunicação entre o R e o MS Excel foi realizada por meio de arquivos .TXT (dados) e .XLSX (resultados) e contou com a interface proporcionada pelo pacote xlsx (Dragulescu, 2014). Adicionalmente, a planilha MS Excel disponibiliza recursos para a impressão de etiquetas com os roteiros a serem percorridos por cada participante, conforme ilustrado pelo botão "gera etiquetas" na Figura 5. Além do moderador, algumas mesas podem contar com um convidado altamente experiente e qualificado (âncora), cujo objetivo consiste em orientar e estimular a troca de ideias. O moderador e o âncora não trocam de mesas.

	A	B	C	D	E	F	G	H
1	ROTEIRIZADOR DE EVENTOS							
2								
3								
4		EXECUTA ALOCAÇÃO			ENTRE COM OS DADOS AQUI			
5					Participantes		106	
6					Âncoras		0	
7		IMPORTA RESULTADO			Pessoas por mesa		10	
8					Número de sessões (se igual a zero, o programa calcula)		0	
9		GERA ETIQUETAS			VALORES CALCULADOS			
10					Nº de mesas		12	
11					Nº de sessões		8	
12		LIMPAR PLANILHA			Nº de arranjos		9	
13					Capacidade total (participantes)		108	
14					Cadeiras ociosas		2	
15					Pessoas por mesa		10	
16					INFORME O NÚMERO EFETIVO DE SESSÕES DO EVENTO		6	
17					Número de encontros por participante e sessão		9	
18					Número de encontros por participante no evento*		27	
19					Total de encontros por sessão*		477	
20					Total encontros no evento*		1,431	
21					*não considera o número de reencontros			

Figura 5: Planilha MS Excel para entrada de dados e visualização de resultados.

A execução do programa é bastante simples. Inicialmente, insira os dados do evento na área indicada em amarelo na Figura 5 e pressione o botão *Executa Alocação* para montar o arquivo de dados em formato texto (dados.txt) e disparar o programa R responsável pela otimização da alocação dos participantes, conforme descrito na macro a seguir, na qual a penúltima linha de código executa o *script R*:

```
Sub executa()
  ChDir ("diretório de instalação do programa")
  Dim ID As Long
  Set wsh = VBA.CreateObject("WScript.Shell")
  ThisFile = "dados.txt"
  If Dir(ThisFile) <> "" Then
    Kill (ThisFile)
  End If
  Open ThisFile For Output As #1
  For j = 1 To 4
    Print #1, Cells(j, 2).Value
  Next j
  Close #1
  ID = wsh.Run("cmd.exe /S /K" & "script_R.Rexec" & "&exit", 1, True)
End Sub
```

A partir dos dados informados no arquivo dados.txt o *script R* define os elementos que formam o PPLI em cada sessão do evento, mais especificamente, o vetor de coeficientes da função objetivo (c), a matriz de coeficientes das restrições (A), o vetor com lado o direito das restrições ou RHS (b), o vetor (d) com os tipos (igualdade e \leq) das restrições e o vetor (tipos) com os tipos de variáveis (binárias “B” e inteiras “I”). A solução do PPLI de cada sessão é realizada no mesmo *script R* por meio da função *Rsymphony_solve_LP* disponível no pacote *Rsymphony*:

```
resposta <- Rsymphony_solve_LP(obj=c, mat=A, dir=d, rhs=b, tipos=tipos)$solution
```

Após a execução do *script R* pressione o botão *Importa Resultado* indicado na Figura 5 para carregar no MS Excel os resultados gerados no R. Entre os resultados importados destacam-se as arranjos dos participantes em cada sessão do evento, os roteiros individuais a serem seguidos por cada participante e o número de reencontros ao longo do evento. Para gerar as etiquetas com os roteiros individuais a serem entregues aos participantes do evento pressione o botão *Gerar Etiquetas*.

4. EXPERIMENTO COMPUTACIONAL

Para ilustrar os resultados gerados pelo aplicativo considere um evento com 106 participantes e 12 mesas com 10 lugares cada uma. Ressalta-se que em cada mesa há um moderador que pertence à organização do evento, assim, cada mesa tem apenas 9 lugares disponíveis para os participantes. O evento é programado para uma duração total de cerca de 2 horas (120 minutos) e cada sessão tem uma duração de cerca de 20 minutos. Assim, as simulações realizadas consideraram um total de 6 seções, ou seja, cada participante percorre 6 mesas.

Na Figura 6 apresentam-se as alocações dos participantes nas 12 mesas ao longo das 6 sessões. Nos quadros ilustrados na Figura 6 são indicadas apenas as 9 cadeiras livres em cada mesa. Vale ressaltar que os resultados gerados pelo programa permitem acomodar até 108 pessoas, logo há duas cadeiras ociosas. Os quadros permitem que os organizadores do evento visualizem rapidamente as composições das mesas ao longo das sessões.

Na Figura 7 tem-se um extrato da tabela contendo os roteiros com a sequências de mesas a serem visitadas pelos 9 participantes que iniciaram na mesa 1. O programa também gera uma planilha com as etiquetas contendo os roteiros que devem ser seguidos pelos participantes do evento. Na Figura 8 tem-se um extrato da planilha com as etiquetas dos três primeiros participantes. Conforme ilustrado na Tabela 1, para um evento com as mesmas dimensões do exemplo os reencontros começam a ocorrer a partir da quarta sessão.

Sessão 1									
	cadeira.1	cadeira.2	cadeira.3	cadeira.4	cadeira.5	cadeira.6	cadeira.7	cadeira.8	cadeira.9
mesa 1	1	2	3	4	5	6	7	8	9
mesa 2	10	11	12	13	14	15	16	17	18
mesa 3	19	20	21	22	23	24	25	26	27
mesa 4	28	29	30	31	32	33	34	35	36
mesa 5	37	38	39	40	41	42	43	44	45
mesa 6	46	47	48	49	50	51	52	53	54
mesa 7	55	56	57	58	59	60	61	62	63
mesa 8	64	65	66	67	68	69	70	71	72
mesa 9	73	74	75	76	77	78	79	80	81
mesa 10	82	83	84	85	86	87	88	89	90
mesa 11	91	92	93	94	95	96	97	98	99
mesa 12	100	101	102	103	104	105	106	107	108

Sessão 2									
	cadeira.1	cadeira.2	cadeira.3	cadeira.4	cadeira.5	cadeira.6	cadeira.7	cadeira.8	cadeira.9
mesa 1	36	45	54	63	72	81	90	97	108
mesa 2	27	44	53	62	71	80	89	96	107
mesa 3	29	43	52	61	70	79	88	95	106
mesa 4	6	18	51	60	69	78	87	94	105
mesa 5	3	17	26	59	68	77	86	93	104
mesa 6	9	16	25	28	67	76	85	92	103
mesa 7	8	15	24	35	42	75	84	91	102
mesa 8	7	14	23	34	38	50	74	83	101
mesa 9	5	13	19	33	41	46	58	82	100
mesa 10	2	10	20	32	37	49	55	66	98
mesa 11	1	12	22	31	40	48	57	65	73
mesa 12	4	11	21	30	39	47	56	64	99

Sessão 3									
	cadeira.1	cadeira.2	cadeira.3	cadeira.4	cadeira.5	cadeira.6	cadeira.7	cadeira.8	cadeira.9
mesa 1	18	19	34	44	48	76	84	93	106
mesa 2	7	29	41	55	72	75	86	92	105
mesa 3	8	17	30	37	53	60	81	83	103
mesa 4	4	12	20	54	58	68	80	95	101
mesa 5	21	36	50	61	69	73	85	91	100
mesa 6	6	10	24	31	38	70	77	99	108
mesa 7	9	13	27	32	43	47	65	87	97
mesa 8	2	16	22	33	39	51	63	89	102
mesa 9	3	15	23	28	40	56	71	90	98
mesa 10	14	25	35	46	57	64	79	96	104
mesa 11	5	11	42	49	59	67	78	88	107
mesa 12	1	26	45	52	62	66	74	82	94

Sessão 4									
	cadeira.1	cadeira.2	cadeira.3	cadeira.4	cadeira.5	cadeira.6	cadeira.7	cadeira.8	cadeira.9
mesa 1	11	23	29	37	51	62	68	96	100
mesa 2	5	26	40	54	60	79	85	99	102
mesa 3	6	41	47	67	73	80	84	98	104
mesa 4	8	10	22	43	59	64	74	90	92
mesa 5	16	27	31	46	56	66	75	83	106
mesa 6	3	13	21	35	44	63	87	95	101
mesa 7	2	12	19	30	45	50	71	78	86
mesa 8	1	18	25	32	53	58	88	91	108
mesa 9	14	36	42	48	55	70	89	94	103
mesa 10	4	28	38	52	65	81	93	105	107
mesa 11	9	15	20	34	39	61	72	77	82
mesa 12	7	17	24	33	49	57	69	76	97

Sessão 5									
	cadeira.1	cadeira.2	cadeira.3	cadeira.4	cadeira.5	cadeira.6	cadeira.7	cadeira.8	cadeira.9
mesa 1	14	15	21	31	43	52	60	80	86
mesa 2	1	24	30	39	46	68	81	88	98
mesa 3	5	45	51	55	64	76	89	91	101
mesa 4	11	25	44	61	71	77	83	97	102
mesa 5	7	10	22	28	47	58	78	82	96
mesa 6	2	17	34	41	56	65	79	94	107
mesa 7	18	26	33	38	66	73	90	95	103
mesa 8	4	13	36	40	49	62	75	92	104
mesa 9	8	20	29	50	57	67	87	93	108
mesa 10	9	12	23	42	53	63	69	99	106
mesa 11	3	16	32	54	70	74	84	100	105
mesa 12	6	19	27	35	37	48	59	72	85

Sessão 6									
	cadeira.1	cadeira.2	cadeira.3	cadeira.4	cadeira.5	cadeira.6	cadeira.7	cadeira.8	cadeira.9
mesa 1	27	32	41	57	70	74	78	85	91
mesa 2	2	48	56	67	69	82	83	95	108
mesa 3	4	13	31	42	50	77	90	96	105
mesa 4	1	15	19	38	47	49	63	79	103
mesa 5	8	33	52	55	71	72	84	99	101
mesa 6	18	20	30	40	59	75	89	97	100
mesa 7	7	16	37	64	80	81	88	94	104
mesa 8	9	12	21	26	29	46	76	98	107
mesa 9	10	25	35	39	54	62	65	86	106
mesa 10	3	11	24	34	45	51	60	73	92
mesa 11	6	17	23	36	43	44	58	66	102
mesa 12	5	14	22	28	53	61	68	87	93

Figura 6: Alocação dos participantes (cada número representa um participante).

Tabela 1: Número acumulado de reencontros .

Sessão	Número de reencontros
1	0
2	0
3	0
4	5
5	22
6	64

participantes	sessão.1	sessão.2	sessão.3	sessão.4	sessão.5	sessão.6
1	1	11	12	8	2	4
2	1	10	8	7	6	2
3	1	5	9	6	11	10
4	1	12	4	10	8	3
5	1	9	11	2	3	12
6	1	4	6	3	12	11
7	1	8	2	12	5	7
8	1	7	3	4	9	5
9	1	6	7	11	10	8

Figura 7: Extrato da tabela de roteiros, cada linha mostra a sequência de mesas a serem visitadas por um participante.

Participante 1		Participante 2		Participante 3	
Sessão 1 - Mesa	1	Sessão 1 - Mesa	1	Sessão 1 - Mesa	1
Sessão 2 - Mesa	11	Sessão 2 - Mesa	10	Sessão 2 - Mesa	5
Sessão 3 - Mesa	12	Sessão 3 - Mesa	8	Sessão 3 - Mesa	9
Sessão 4 - Mesa	8	Sessão 4 - Mesa	7	Sessão 4 - Mesa	6
Sessão 5 - Mesa	2	Sessão 5 - Mesa	6	Sessão 5 - Mesa	11
Sessão 6 - Mesa	4	Sessão 6 - Mesa	2	Sessão 6 - Mesa	10

Figura 8: Exemplos de etiquetas distribuídas aos participantes.

5. CONCLUSÕES

Apresentou-se um modelo de Programação Linear Inteira para organizar fóruns empresariais. O modelo proposto visa minimizar o número de reencontros de forma a contribuir para o maior número de contatos entre os participantes. O modelo foi implementado em R e uma planilha MS Excel serve de interface com o usuário. O aplicativo já foi utilizado em quatro eventos e a experiência nestas oportunidades tem recebido elogios dos participantes. Além de reduzir o número de reencontros, o aplicativo agilizou a organização dos fóruns em que foi utilizado. Os resultados obtidos mostram que a metodologia proposta é promissora, porém mais investigações devem ser realizadas no sentido de ampliar a capacidade do programa visando a sua aplicação na organização de eventos com um grande número de participantes.

6. REFERÊNCIAS

- BARTLETT, A.; CHARTIER, T.P.; LANGVILLE, A.N.; RANKIN, T.D.** An Integer Programming Model for the Sudoku Problem, *Convergence*, v. 8, May, 2008.
- BELLOWS, M. L. & PETERSON, D.L.** Finding an optimal seating chart, *Annals of Improbable Research*, February, 2012.
- DRAGULESCU, A. A.** xlsx: Read, write, format Excel 2007 and Excel 97/2000/XP/2003 files. R package version 0.5.7., 2014. Disponível em: < <http://CRAN.R-project.org/package=xlsx> >. Acesso em: 20 jan. 2017.
- HARTER, R.; HORNIK, K.; THEUSSL, L.; SZYMANSKI, C.** Rsymphony: SYMPHONY in R. R package version 0.1-22, 2016. Disponível em: < <http://CRAN.R-project.org/package=Rsymphony> >. Acesso em: 20 jan. 2017.
- JELLEN, B. & SYRSTAD, T.** VBA e Macros para Microsoft Office Excel 2007, Prentice Hall do Brasil, 2008.
- JOHNSON, S.** De onde vêm as boas ideias. Rio de Janeiro: Zahar, 2011.
- MEISSNER, P.** Making R Files Executable (under Windows). R-Bloggers, Feb., 2015. Disponível em: < <https://www.r-bloggers.com/making-r-files-executable-under-windows/> >. Acesso em: 20 jan. 2016.

R CORE Team. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria, 2014. Disponível em: <<https://www.R-project.org/>>. Acesso em: 20 jan. 2017.

RAGSDALE, C. T. Spreadsheet Modeling & Decision Analysis: A Practical Introduction to Management Science, 4. ed. Mason: Southern-Western, 2004.