Unrelated Parallel-machine Scheduling Problems with General Truncated Job-dependent Learning Effect

Ji-Bo Wang¹, Chou-Jung Hsu².*

¹School of Science, Shenyang Aerospace University, Shenyang 110136, China;
²*Department of Industrial Management, Nan Kai University of Technology, Nantou 542, Taiwan, R.O.C

Email: ¹wangjibo75@163.com, ²*jrsheu@nkut.edu.tw

Copyright © 2014 by author(s) and Scientific Research Publishing Inc.
This work is licensed under the Creative Commons Attribution International License (CC BY).
http://creativecommons.org/licenses/by/4.0/

Abstract

In this paper, we consider scheduling problems with general truncated job-dependent learning effect on unrelated parallel-machine. The objective functions are to minimize total machine load, total completion (waiting) time, total absolute differences in completion (waiting) times respectively. If the number of machines is fixed, these problems can be solved in $O(n^{m+2})$ time respectively, where $m$ is the number of machines and $n$ is the number of jobs.

Keywords

Scheduling; Unrelated Parallel Machines; Truncated Job-dependent Learning

1. Introduction

In modern planning and scheduling problems, there are many real situations where the processing time of jobs may be subject to change due to learning effect. An extensive survey of different scheduling models and problems with learning effects could be found in Biskup [1]. More recently, Janiak et al. [2] studied a single processor problem with a S-shaped learning model. They proved that the makespan minimization problem is strongly NP-hard. Lee [3] considered scheduling jobs with general position-based learning curves. For some single machine and a two-machine flowshop scheduling problems, they presented the optimal solution respectively. Lee [4] considered single-machine scheduling jobs with general learning effect and past-sequence-dependent setup time. For some single machine scheduling problems, they presented the optimal solution respectively. Lee and Wu [5], and Wu and Lee [6] considered scheduling jobs with learning effects. They proved that some single
machine and flowshop scheduling problems can be solved in polynomial time respectively. Lee et al. [7] considered a single-machine scheduling problem with release times and learning effect. Lee et al. [8] considered a makespan minimization uniform parallel-machine scheduling problem with position-based learning curves. Lee and Chung [9], Sun et al. [10, 11], and Wang et al. [12] considered flow shop scheduling with learning effects. Wu et al. [13], Wu et al. [14], Wu et al. [15] and Wang et al. [16] considered scheduling problems with the truncated learning effect.

Recently, Wang et al. [17] considered several scheduling problems on a single machine with truncated job-dependent learning effect, i.e., the actual processing time of job $J_j$ is $p_{ij}^b = p_j \max \{r^j, b\}$ if it is scheduled in the $r$th position of a sequence, where $a_j \leq 0$ is the job-dependent learning index of job $J_j$, and $b$ is a truncation parameter with $0 < b < 1$. In this paper, we study scheduling problems with general truncated job-dependent learning effect on unrelated parallel-machine. The objective is to minimize total machine load, total completion (waiting) time, total absolute differences in completion (waiting) times respectively.

**2. Problems Description**

There are $n$ independent jobs $N = \{J_1,J_2,\cdots,J_n\}$ to be processed on $m$ unrelated paralle-machine $M = \{M_1,M_2,\cdots,M_m\}$. Let $(n_1,n_2,\cdots,n_m)$ denote a job-allocation vector, where $n_i$ denotes the number of jobs assigned to machine $M_i$, and $\sum_{i=1}^{m} n_i = n$. In this paper, we assume that the actual processing time of job $J_j$ scheduled on machine $M_i$ is

$$p_{ij}^b = p_j \max \{f_j(r),b\}, \quad i = 1,2,\ldots,m; \quad r = 1,2,\ldots,n,$$

where $p_j \geq 0$ denotes the normal (basic) processing time of job $J_j$ ($j = 1,2,\ldots,n$) on machine $M_i$, $r$ is the position of a sequence, $b$ is a truncation parameter with $0 < b < 1$, $f_j(r)$ is the general case of positional learning for job $J_j$ on machine $M_i$, special $f_j(r) = r^k$ is the polynomial learning index for job $J_j$ on machine $M_i$ ($a_j < 0$), $f_j(r) = b^t$ is the exponential learning index for job $J_j$ on machine $M_i$ ($0 < b < 1$).

Let $C_{ij}$ and $W_{ij} = C_{ij} - p_j$ be the completion and waiting time for job $J_j$ on machine $M_i$, respectively. The goal is to determine the jobs assigned to corresponding each machine and the corresponding optimal schedule so that the following objective functions is to be minimized: the total machine load $\sum_{i=1}^{m} C_{ij}$, the total completion (waiting) times $\sum_{i=1}^{m} \sum_{j=1}^{n_i} C_{ij} \left( \sum_{i=1}^{m} \sum_{j=1}^{n_i} W_{ij} \right)$, the total absolute differences in completion (waiting) times $\sum_{i=1}^{m} \sum_{j=1}^{n_i} \sum_{k=1}^{n_i} |C_{ik} - C_{ij}| \left( \sum_{i=1}^{m} \sum_{j=1}^{n_i} \sum_{k=1}^{n_i} W_{ik} - W_{ij} \right)$, where $C_{ij}$ denotes the makespan of machine $M_i$.

Using the three-field notation [18] the problems can be denoted as $Rm|Y|Z$, where $Y$ denote the model (1), $Z \in \{\sum_{i=1}^{m} C_{ij}, \sum_{i=1}^{m} \sum_{j=1}^{n_i} C_{ij}, \sum_{i=1}^{m} \sum_{j=1}^{n_i} W_{ij}, \sum_{i=1}^{m} \sum_{j=1}^{n_i} \sum_{k=1}^{n_i} |C_{ik} - C_{ij}| \sum_{i=1}^{m} \sum_{j=1}^{n_i} \sum_{k=1}^{n_i} W_{ik} - W_{ij} \}$.

**3. Main results**

Let $p_j$ denote the actual processing time of a job when it is scheduled in position $j$ on machine $M_i$, then $f_{i,j}(r)$, $C_{i,j}$, $W_{i,j}$ are defined similarly.

**Lemma 1.** For a given permutation $\pi = (J_{i_1};J_{i_2};\ldots;J_{i_{n_i}})$ on machine $M_i$,

$$\sum_{i=1}^{n_i} C_{i,j} = \sum_{i=1}^{n_i} \sum_{j=1}^{n_i} p_{i,j} \max \{f_{i,j}(r),b\}$$

$$\sum_{i=1}^{n_i} \sum_{j=1}^{n_i} C_{ij} = \sum_{i=1}^{n_i} \sum_{j=1}^{n_i} (n_j + 1) p_{i,j} \max \{f_{i,j}(r),b\}$$

$$\sum_{i=1}^{n_i} \sum_{j=1}^{n_i} W_{ij} = \sum_{i=1}^{n_i} \sum_{j=1}^{n_i} (n_j - j) p_{i,j} \max \{f_{i,j}(r),b\}$$

Special description of the title. (dispensable)
If the vector \((n_1, n_2, \ldots, n_n)\) is given, let \(X_{ir}\) be a 0/1 variable such that \(X_{ir} = 1\) if job \(J_i\) (\(j = 1, 2, \ldots, n\)) is assigned at position \(r\) (\(r = 1, 2, \ldots, n\)) on machine \(M_i\) (\(i = 1, 2, \ldots, m\)), and \(X_{ir} = 0\), otherwise. Then, the problem \(Rm[Y]Z\) (where \(Rm[Y]Z = \sum_{i=1}^{m} \sum_{j=1}^{n} (C_{ij} - C_{yj}) + \sum_{i=1}^{m} \sum_{j=1}^{n} W_{ia} - W_{ij}\)) can be solved by the following assignment problem:

\[
\min Z = \sum_{i=1}^{m} \sum_{j=1}^{n} \lambda_{ir} p_{ij} \max \{f_{ij}(r), b\} X_{ijr}
\]

Subject to:

\[
\sum_{i=1}^{m} \sum_{j=1}^{n} X_{ijr} = 1, j = 1, 2, \ldots, n,
\]

\[
\sum_{j=1}^{n} X_{ijr} = 1, i = 1, 2, \ldots, m, r = 1, 2, \ldots, n,
\]

\[
X_{ijr} = 0, \text{ or } 1, j = 1, 2, \ldots, n, i = 1, 2, \ldots, m, r = 1, 2, \ldots, n,
\]

where \(\lambda_{ir} = 1\) for \(\sum_{i=1}^{m} C_{ij} + \lambda_{ir} = (n_j + 1)\) for \(\sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij}\), \(\lambda_{ir} = (n_j - r)\) for \(\sum_{i=1}^{m} \sum_{j=1}^{n} W_{ai}\) , \(\lambda_{ir} = (r - 1)(n_j - r + 1)\) for \(\sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{m} C_{ik} - C_{yj}\), \(\lambda_{ir} = r(n_j - j)\) for \(\sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{m} W_{ia} - W_{ij}\).

Now, the question is how many vectors \((n_1, n_2, \ldots, n_n)\) exist. Obviously \(n_i\) may be 0, 1, 2, \ldots, \(n\). So if the numbers of jobs assigned to the first \(m-1\) machines is given, the number of jobs assigned to the last machine is then determined uniquely \((\sum_{i=1}^{m} n_i = n)\). Therefore, the upper bound of \((n_1, n_2, \ldots, n_n)\) is \((n+1)^{m-1}\). Based on the above analysis, we have the following result.

**Theorem 1.** For a given constant \(m\), \(Rm[Y]Z\) can be solved in \(O(n^{m+2})\) time, where

\[
Z \in \left\{ \sum_{i=1}^{m} C_{i} + \sum_{j=1}^{n} C_{yj}, \sum_{j=1}^{n} W_{ij}, \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{m} (C_{ik} - C_{yj}), \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{m} W_{ia} - W_{ij} \right\}.
\]

**Proof.** As discussed above, to solve the problem \(Rm[Y]Z\), polynomial number (i.e., \((n+1)^{m-1}\)) of assignment problems need to be solved. Each assignment problem is solved in \(O(n^3)\) time (by using the Hungarian method). Hence, the time complexity of the problem \(Rm[Y]Z\) can be solved in \(O(n^{m+2})\) time, where

\[
Z \in \left\{ \sum_{i=1}^{m} C_{i}, \sum_{j=1}^{n} C_{yj}, \sum_{j=1}^{n} W_{ij}, \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{m} C_{ik} - C_{yj}, \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{m} W_{ia} - W_{ij} \right\}.
\]

Note that if the number of machines \(m\) is fixed, then the problem \(Rm[Y]Z\) can be solved in polynomial time.

Based on the above analysis, we can determine the optimal solution for the problem \(Rm[Y]Z\) via the following algorithm:

**Algorithm 1**

*Step 1.* For each possible vector \((n_1, n_2, \ldots, n_n)\), solve the assignment problem (2)-(5). Then, obtain the optimal schedule and the corresponding objective function \(Z\).

*Step 2.* The optimal solution for the problem is the one with the minimum value of the objective function \(Z\), where \(Z \in \left\{ \sum_{i=1}^{m} C_{i} + \sum_{j=1}^{n} C_{yj}, \sum_{j=1}^{n} W_{ij}, \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{m} (C_{ik} - C_{yj}), \sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{m} W_{ia} - W_{ij} \right\}.

The following example illustrates the working of Algorithm 1 to find the optimal solution for the problem \(Rm[Y]\sum_{i=1}^{m} C_{yj}.

**Example 1.** There are \(n = 5\) jobs and \(f_{ij}(r) = r^{5i}\). The number of machines is \(m = 2\) and \(p_{11} = 15\), \(p_{12} = 11\),
Solution. When \( n_1 = 0, n_2 = 5 \), the positional weights on machine \( M_2 \) are \( \theta_{21} = 5 \), \( \theta_{22} = 4 \), \( \theta_{23} = 3 \), \( \theta_{24} = 2 \), \( \theta_{25} = 1 \). Then values \( \theta_{ir} p_{ij} \max \{r_{ij}, b\} \) are given in Table 1 (the bold value is the optimal solution of the assignment problem (2)-(5)).

We solve the assignment problem (2)-(5) to \( \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij} = 339.65119 \).

When \( n_1 = 1, n_2 = 4 \), the positional weights on machine \( M_1 \) and \( M_2 \) are \( \theta_{11} = 1 \), \( \theta_{21} = 4 \), \( \theta_{22} = 3 \), \( \theta_{23} = 2 \), \( \theta_{24} = 1 \). Then values \( \theta_{ir} p_{ij} \max \{r_{ij}, b\} \) are given in Table 2. We solve the assignment problem (2)-(5) to obtain that the optimal schedule on machine \( M_1 \) is \( [J_1, J_2] \), and on machine \( M_2 \) is \( [J_3, J_4, J_2, J_1] \).

The objective function is \( \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij} = 81.05875 \).

When \( n_1 = 2, n_2 = 3 \), the positional weights on machine \( M_1 \) and \( M_2 \) are \( \theta_{11} = 2 \), \( \theta_{12} = 1 \), \( \theta_{13} = 3 \), \( \theta_{22} = 2 \), \( \theta_{23} = 1 \). Then values \( \theta_{ir} p_{ij} \max \{r_{ij}, b\} \) are given in Table 3. We solve the assignment problem (2)-(5) to obtain that the optimal schedule on machine \( M_1 \) is \( [J_1, J_2] \), and on machine \( M_2 \) is \( [J_3, J_4, J_1] \).

The objective function is \( \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij} = 61.77443 \).

When \( n_1 = 3, n_2 = 2 \), the positional weights on machine \( M_1 \) and \( M_2 \) are \( \theta_{11} = 3 \), \( \theta_{12} = 2 \), \( \theta_{13} = 1 \), \( \theta_{21} = 2 \), \( \theta_{22} = 1 \). Then values \( \theta_{ir} p_{ij} \max \{r_{ij}, b\} \) are given in Table 4. We solve the assignment problem (2)-(5) to obtain that the optimal schedule on machine \( M_1 \) is \( [J_3, J_4, J_2] \), and on machine \( M_2 \) is \( [J_1, J_3] \).

The objective function is \( \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij} = 59.38394 \).

When \( n_1 = 4, n_2 = 1 \), the positional weights on machine \( M_1 \) and \( M_2 \) are \( \theta_{11} = 4 \), \( \theta_{12} = 3 \), \( \theta_{13} = 2 \), \( \theta_{14} = 1 \), \( \theta_{21} = 1 \). Then values \( \theta_{ir} p_{ij} \max \{r_{ij}, b\} \) are given in Table 5. We solve the assignment problem (2)-(5) to obtain that the optimal schedule on machine \( M_1 \) is \( [J_1, J_4, J_2, J_3] \), and on machine \( M_2 \) is \( [J_2] \).

The objective function is \( \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij} = 69.93119 \).

When \( n_1 = 5, n_2 = 0 \), the positional weights on machine \( M_1 \) and \( M_2 \) are \( \theta_{11} = 5 \), \( \theta_{12} = 4 \), \( \theta_{13} = 3 \), \( \theta_{14} = 2 \), \( \theta_{15} = 1 \). Then values \( \theta_{ir} p_{ij} \max \{r_{ij}, b\} \) are given in Table 6. We solve the assignment problem (2)-(5) to obtain that the optimal schedule on machine \( M_1 \) is \( [J_1, J_4, J_2, J_3, J_1] \). The objective function is \( \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij} = 98.58071 \).

Hence, the optimal schedule on machine \( M_1 \) is \( [J_4, J_5, J_2] \), and on machine \( M_2 \) is \( [J_3, J_1] \). The optimal objective function is \( \sum_{i=1}^{m} \sum_{j=1}^{n} C_{ij} = 59.38394 \).
Table 1. The $\theta_{i,p_i} \max \{ r^*, b \}$ values of Example 1 for $n_1 = 0, n_2 = 5$

<table>
<thead>
<tr>
<th>$ij \backslash ir$</th>
<th>$\theta_{21}$</th>
<th>$\theta_{22}$</th>
<th>$\theta_{23}$</th>
<th>$\theta_{24}$</th>
<th>$\theta_{25}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J_{21}$</td>
<td>60</td>
<td>38.45135</td>
<td>25.32933</td>
<td>16.80000</td>
<td><strong>8.40000</strong></td>
</tr>
<tr>
<td>$J_{22}$</td>
<td>50</td>
<td>34.58149</td>
<td>23.81912</td>
<td><strong>14.94849</strong></td>
<td>7.13208</td>
</tr>
<tr>
<td>$J_{23}$</td>
<td>45</td>
<td>29.03910</td>
<td><strong>19.20685</strong></td>
<td>12.60000</td>
<td>6.30000</td>
</tr>
<tr>
<td>$J_{24}$</td>
<td>90</td>
<td>54.19170</td>
<td>36.87501</td>
<td>22.94328</td>
<td>11.20000</td>
</tr>
<tr>
<td>$J_{25}$</td>
<td>40</td>
<td><strong>27.09585</strong></td>
<td>18.43750</td>
<td>11.20000</td>
<td>5.60000</td>
</tr>
</tbody>
</table>

Table 2. The $\theta_{i,p_i} \max \{ r^*, b \}$ values of Example 1 for $n_1 = 1, n_2 = 4$

<table>
<thead>
<tr>
<th>$ij \backslash ir$</th>
<th>$\theta_{21}$</th>
<th>$\theta_{22}$</th>
<th>$\theta_{23}$</th>
<th>$\theta_{24}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J_{11}$</td>
<td>15</td>
<td>48</td>
<td>28.83852</td>
<td>16.88622</td>
</tr>
<tr>
<td>$J_{12}$</td>
<td>11</td>
<td>40</td>
<td>25.93612</td>
<td><strong>15.87942</strong></td>
</tr>
<tr>
<td>$J_{13}$</td>
<td>14</td>
<td>36</td>
<td><strong>21.77933</strong></td>
<td>12.80457</td>
</tr>
<tr>
<td>$J_{14}$</td>
<td>3</td>
<td>64</td>
<td>40.64377</td>
<td>24.58334</td>
</tr>
<tr>
<td>$J_{15}$</td>
<td>9</td>
<td><strong>32</strong></td>
<td>19.62965</td>
<td>11.63469</td>
</tr>
</tbody>
</table>

Table 3. The $\theta_{i,p_i} \max \{ r^*, b \}$ values of Example 1 for $n_1 = 2, n_2 = 3$

<table>
<thead>
<tr>
<th>$ij \backslash ir$</th>
<th>$\theta_{21}$</th>
<th>$\theta_{22}$</th>
<th>$\theta_{23}$</th>
<th>$\theta_{24}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J_{11}$</td>
<td>30</td>
<td>12.78952</td>
<td>36</td>
<td>19.22568</td>
</tr>
<tr>
<td>$J_{12}$</td>
<td>22</td>
<td><strong>8.81177</strong></td>
<td>30</td>
<td>17.29074</td>
</tr>
<tr>
<td>$J_{13}$</td>
<td>28</td>
<td>11.77255</td>
<td>27</td>
<td><strong>14.51955</strong></td>
</tr>
<tr>
<td>$J_{14}$</td>
<td>6</td>
<td>2.35375</td>
<td>48</td>
<td>27.09585</td>
</tr>
<tr>
<td>$J_{15}$</td>
<td>18</td>
<td>7.51579</td>
<td><strong>24</strong></td>
<td>13.08643</td>
</tr>
</tbody>
</table>

Table 4. The $\theta_{i,p_i} \max \{ r^*, b \}$ values of Example 1 for $n_1 = 3, n_2 = 2$

<table>
<thead>
<tr>
<th>$ij \backslash ir$</th>
<th>$\theta_{21}$</th>
<th>$\theta_{22}$</th>
<th>$\theta_{23}$</th>
<th>$\theta_{24}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J_{11}$</td>
<td>45</td>
<td>25.57905</td>
<td>11.65074</td>
<td>24</td>
</tr>
<tr>
<td>$J_{12}$</td>
<td>33</td>
<td>17.62354</td>
<td><strong>7.739517</strong></td>
<td>20</td>
</tr>
<tr>
<td>$J_{13}$</td>
<td>42</td>
<td>23.54510</td>
<td>10.63770</td>
<td><strong>18</strong></td>
</tr>
<tr>
<td>$J_{14}$</td>
<td>9</td>
<td>4.70751</td>
<td>2.10000</td>
<td>32</td>
</tr>
<tr>
<td>$J_{15}$</td>
<td>27</td>
<td><strong>15.03158</strong></td>
<td>6.76380</td>
<td>16</td>
</tr>
</tbody>
</table>
Table 5. The $\theta_{ij} p_{ir} \max \{r^x, b\}$ values of Example 1 for $n_1 = 4, n_2 = 1$

<table>
<thead>
<tr>
<th>$ij \backslash ir$</th>
<th>$\theta_{11}$</th>
<th>$\theta_{21}$</th>
<th>$\theta_{22}$</th>
<th>$\theta_{23}$</th>
<th>$\theta_{24}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J_{11}$</td>
<td>60</td>
<td>38.36857</td>
<td>23.30147</td>
<td><strong>10.90479</strong></td>
<td>12</td>
</tr>
<tr>
<td>$J_{12}$</td>
<td>44</td>
<td>26.43531</td>
<td><strong>15.47903</strong></td>
<td>7.70000</td>
<td>10</td>
</tr>
<tr>
<td>$J_{13}$</td>
<td>56</td>
<td>35.31765</td>
<td>21.2754</td>
<td>9.89950</td>
<td>9</td>
</tr>
<tr>
<td>$J_{14}$</td>
<td>12</td>
<td>7.06126</td>
<td>4.20000</td>
<td>2.10000</td>
<td>16</td>
</tr>
<tr>
<td>$J_{15}$</td>
<td>36</td>
<td><strong>22.54737</strong></td>
<td>13.52761</td>
<td>6.30000</td>
<td>8</td>
</tr>
</tbody>
</table>

Table 6. The $\theta_{ij} p_{ir} \max \{r^x, b\}$ values of Example 1 for $n_1 = 5, n_2 = 0$

<table>
<thead>
<tr>
<th>$ij \backslash ir$</th>
<th>$\theta_{21}$</th>
<th>$\theta_{22}$</th>
<th>$\theta_{23}$</th>
<th>$\theta_{24}$</th>
<th>$\theta_{25}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$J_{11}$</td>
<td>75</td>
<td>51.15809</td>
<td>34.95221</td>
<td>21.80959</td>
<td><strong>10.50000</strong></td>
</tr>
<tr>
<td>$J_{12}$</td>
<td>55</td>
<td>35.24707</td>
<td><strong>23.21855</strong></td>
<td>15.40000</td>
<td>7.70000</td>
</tr>
<tr>
<td>$J_{13}$</td>
<td>70</td>
<td>47.09020</td>
<td>31.91310</td>
<td><strong>19.79899</strong></td>
<td>9.80000</td>
</tr>
<tr>
<td>$J_{14}$</td>
<td><strong>15</strong></td>
<td>9.41501</td>
<td>6.30000</td>
<td>4.20000</td>
<td>2.10000</td>
</tr>
<tr>
<td>$J_{15}$</td>
<td>45</td>
<td><strong>30.06317</strong></td>
<td>20.29141</td>
<td>12.60000</td>
<td>6.30000</td>
</tr>
</tbody>
</table>

Acknowledgements

Hsu was supported by the Ministry Science and Technology of Taiwan under Grant MOST 104-2221-E-252-002-MY2.

References


http://www.palgrave-journals.com/jors/journal/v64/n1/abs/jors201246a.html