On speeding convergence of an iterative eigenvalue process

By Ivan Erdelyi*

This paper describes a practical procedure suitable for accelerating convergence to the dominant latent root of matrices reducible to diagonal form. The adaptation of the technique for an automatic computer requires a very limited working store. Some examples illustrate the method, and certain particularities are discussed.

This paper is concerned with the problem of accelerating the time-consuming iterations for the determination of the dominant latent root and vector of a given (n, n) matrix.

The direct iteration consists in repeated premultiplications of a current vector y_k by the given matrix A:

$$y_k = Ay_{k-1}, \quad y_0 \text{ arbitrary}, \quad k = 1, 2, \ldots \tag{1}$$

Since the initial vector y_0 of the foregoing scheme may be expanded

$$y_0 = \sum_{i=1}^{n} c_i x_i$$

in terms of the latent vectors $x_1, x_2, \ldots, x_n$ with the coefficients $c_1, c_2, \ldots, c_n$ not all zero, the kth iterate (1) assumes the form

$$y_k = \sum_{i=1}^{n} c_i \lambda_i^k x_i$$

where $\lambda_1, \lambda_2, \ldots, \lambda_n$ denote the respective latent roots. Further assume

$$|\lambda_1| > |\lambda_2| \geqslant \ldots \geqslant |\lambda_n|.$$

Supposing c_1 non-zero, the kth iterate

$$y_k = \lambda_1^k \left[c_1 x_1 + \sum_{i=2}^{n} c_i \left(\frac{\lambda_i}{\lambda_1}\right)^k x_i \right] \tag{1'}$$

tends to x_1 with the rate of convergence governed by

$$r = \lambda_2/\lambda_1. \tag{2}$$

A slow convergence is the consequence of an unfavourable quotient r. The iterative process applied to the modified matrix $A - qI$, where I is the unit matrix and q a conveniently chosen scalar, converges more rapidly if

$$\left|\frac{\lambda_2 - q}{\lambda_1 - q}\right| < r.$$

The best choice of q is given by

$$q = \tfrac{1}{2}(\lambda_2 + \lambda_n).$$

However, neither λ_2 nor λ_n is a priori known, so that there is an inherent uncertainty concerning a good guess for q. In fact, it is not certain that acceleration in general will have any significant advantage, since for best application we need some knowledge of the distribution of the roots, and this is often lacking in practical problems.

The procedure described below extracts some useful information about the distribution of the appropriate roots from the behaviour of the successive y_k.

This information is then used for obtaining suitable values for q which may improve the rate of convergence.

It should be remarked that iterations with $(A - qI)$ may converge to a solution other than the largest root. As a matter of fact, when operating with the modified matrix $(A - qI)$, the smallest root λ_n may become dominant. A similar case must then be ascertained by further computation.

Determination of the accelerating parameter q

The process may start at a stage of the iteration when the iterate reduced to $x_1 + \varepsilon x_2$ satisfies the following conditions:

(1) the largest component of $x_1 + \varepsilon x_2$ is in the same position j as that of x_1;

(2) the powers of ε higher than the first can be neglected. For definiteness assume the latent vectors normalized so that their largest component is unity, and denote by $p(|p| \leqslant 1)$ the jth component of x_2.

A normalized iterate can then assume the form

$$y = \frac{x_1 + \varepsilon x_2}{1 + \varepsilon p}.$$

The next three iterations give

$$Ay_{i-1} = M_i y_i, \quad i = 1, 2, 3,$$

where M_i are the normalizing factors, and hence

$$M_i = \lambda_1 \frac{1 + \varepsilon p r^i}{1 + \varepsilon p r^{i-1}}$$

$$y_i = \frac{x_1 + \varepsilon r^i x_2}{1 + \varepsilon p r^i}, \quad i = 1, 2, 3.$$

The foregoing results may be collected into the following non-linear system:

$$\left.\begin{aligned} M_1 &= \lambda_1(1 - \varepsilon p + \varepsilon p r) + O(\varepsilon^2) \\ M_2 &= \lambda_1(1 - \varepsilon p r + \varepsilon p r^2) + O(\varepsilon^2) \\ M_3 &= \lambda_1(1 - \varepsilon p r^2 + \varepsilon p r^3) + O(\varepsilon^2). \end{aligned}\right\} \tag{3}$$

* *Mathematical Methods Department, Olivetti-General Electric, Milan, Italy.*

Ignoring the powers of ε higher than the first, a simple calculation gives

$$r = \frac{M_3 - M_2}{M_2 - M_1} \tag{4}$$

$$\lambda_2 = \lambda_1 r = \frac{r}{1-r}(M_2 - M_1 r). \tag{5}$$

Solutions (4), (5) may sometimes be very crude approximations; nevertheless, they are good enough to reduce the ratio r to the fraction

$$\frac{\lambda_2 - q}{\lambda_1 - q} = \frac{r}{m} \tag{6}$$

where, within the limits of applicability of this technique, a value of 2, 3 or 4 may be taken for m.

From (6) we derive the accelerating parameter q as follows:

$$q = \frac{m-1}{m-r}\lambda_2 \tag{7}$$

with r and λ_2 given by relations (4) and (5).

It is easy now to verify the extreme cases

$$\begin{aligned} m &= 1, \quad q = 0; \\ r &= 1, \quad q = \lambda_1 = \lambda_2; \\ r &= 0, \quad q = 0, \end{aligned}$$

for formulae (4), (5) and (7).

Efficiency

When considering the efficiency of this method it is worthwhile to examine how the foregoing results vary as the effect of the third latent vector x_3 is considered.

The normalized iterate may then be expressed in terms of the first three latent vectors x_1, x_2, x_3 in the form

$$y = \frac{x_1 + \varepsilon x_2 + \eta x_3}{1 + \varepsilon p + \eta t}$$

where the small coefficient η has to weight the contribution of x_3, and $t(|t| \leqslant 1)$ denotes the jth component of the latter.

By a calculation similar to that of the foregoing section, new factors N_1, N_2, N_3 may be determined, in terms of the old factors M_1, M_2, M_3. They are given by

$$\left.\begin{aligned} N_1 &= M_1 + \eta t(s-1)\lambda_1 \\ N_2 &= M_2 + \eta t s(s-1)\lambda_1 \\ N_3 &= M_3 + \eta t s^2(s-1)\lambda_1 \end{aligned}\right\} \tag{8}$$

where

$$s = \lambda_3/\lambda_1.$$

A new approximation r' for the ratio (2) is given by

$$r' = \frac{N_3 - N_2}{N_2 - N_1}.$$

The replacement of N_1, N_2, N_3 by (8), in accordance with relations (3), (4) and under the condition

$$\left|\frac{\eta t}{\varepsilon p}\right| < \left|\frac{1-r}{1-s}\right|^2 \tag{9}$$

leads to

$$r' = r + \frac{\eta t}{\varepsilon p}(s-r)\left(\frac{1-s}{1-r}\right)^2 + \mathrm{O}\left[\frac{\eta t}{\varepsilon p}\left(\frac{1-s}{1-r}\right)^2\right]^2. \tag{10}$$

When p is zero or very near to it this method is no longer efficient, but in all other cases it is always possible to reach a stage of iteration when condition (9) is satisfied.

For the legitimate use of this procedure it is perhaps reasonable to show that, admitting (9), the error in the determination of the accelerating parameter q due to the contribution of the third latent vector x_3 does not exceed the order of magnitude of the quantity $|\eta t/\varepsilon p|$.

In fact, the increment Δq of the parameter q may be expressed in terms of the variations Δr, ΔM_1, ΔM_2 of r, M_1, M_2, respectively. From relations (10) and (8) we have

$$\left.\begin{aligned} \Delta r &= r' - r = \frac{\eta t}{\varepsilon p}(s-r)\left(\frac{1-s}{1-r}\right)^2 \\ \Delta M_1 &= N_1 - M_1 = \eta t(s-1)\lambda_1 \\ \Delta M_2 &= N_2 - M_2 = \eta t s(s-1)\lambda_1. \end{aligned}\right\} \tag{11}$$

The relative error may then be expressed by the expansion.

$$\frac{\Delta q}{q} \cong \left[\frac{1}{m-r} + \frac{1}{r(1-r)} - \frac{M_1}{M_2 - M_1 r}\right] (s-r)\left(\frac{1-s}{1-r}\right)^2 \frac{\eta t}{\varepsilon p} + \frac{s-r}{M_2 - M_1 r}\lambda_1(s-1)\eta t,$$

confining ourselves to the first-order variation

$$\frac{\Delta q}{q} \cong \frac{\partial \log q}{\partial r}\Delta r + \frac{\partial \log q}{\partial M_1}\Delta M_1 + \frac{\partial \log q}{\partial M_2}\Delta M_2$$

and using relations (4), (5), (7) and (11).

It is now clear that the dominant part of the foregoing expression is due to that of the Δr, and its order of magnitude is $|\eta t/\varepsilon p|$.

It will be pointed out below in commenting on some examples that a limited influence of some distant latent roots and vectors may favourably influence, in certain cases, the determination of q.

Examples

Three examples testing the acceleration procedure are given. They start with a (24, 24) sparse matrix **(Table 1)**. In order to derive two new (24, 24) matrices **(Tables 2–3)** deflations preserving matrix dimensions* are used. Computational results according with an output criterion

* The method of deflation uses the latent vectors associated with the dominant root both of the given matrix and of its transpose (Wilkinson (1954), p. 546).

Table 1

						.0	.0	.0	-1.5	1.0	1.0				.0	.0	.0	1.5	.5	.5	.0	.0	.0
						.0	.0	.0	4.5	-3.0	-1.0				.0	.0	.0	-1.5	-.5	-.5	-3.0	-1.0	.0
		0				.0	.0	.0	9.0	-4.5	-1.5		0		.0	.0	.0	-3.0	-1.5	-1.5	-6.0	-3.0	.0
						.0	.0	2.0	.0	.0	.0				1.5	.5	.5	.0	.0	.0	-1.5	-.5	.5
						9.0	-6.0	.0	.0	.0	.0				-1.5	-.5	-.5	-6.0	-2.0	.0	-1.5	-.5	.5
						18.0	-9.0	.0	.0	.0	.0				-3.0	-1.5	-1.5	-12.0	-6.0	.0	-3.0	-1.5	1.5
.0	.0	.0	.0	.0	2.0							1.5	.5	.5	.0	.0	.0	-1.5	-.5	.5			
.0	.0	.0	9.0	-6.0	.0							-1.5	-.5	-.5	-6.0	-2.0	.0	-1.5	-.5	.5			
.0	.0	.0	18.0	-9.0	.0			0				-3.0	-1.5	-1.5	-12.0	-6.0	.0	-3.0	-1.5	1.5		0	
1.5	-1.0	1.0	.0	.0	.0							.0	.0	.0	-1.5	-.5	.5	.0	.0	.0			
4.5	-3.0	1.0	.0	.0	.0							-3.0	-1.0	.0	-1.5	-.5	.5	.0	.0	.0			
9.0	-4.5	1.5	.0	.0	.0							-6.0	-3.0	.0	-3.0	-1.5	1.5	.0	.0	.0			
			.0	.0	.0	-1.5	.5	-.5	.0	.0	.0							.0	.0	.0	1.5	1.0	-1.0
			.0	.0	.0	-1.5	.5	-.5	-3.0	1.0	.0							.0	.0	.0	4.5	3.0	-1.0
	0		.0	.0	.0	3.0	-1.5	1.5	6.0	-3.0	.0			0				.0	.0	.0	-9.0	-4.5	1.5
			-1.5	.5	-.5	.0	.0	.0	1.5	-.5	-.5							.0	.0	-2.0	.0	.0	.0
			-1.5	.0	-.5	-6.0	2.0	.0	-1.5	.5	.5							9.0	6.0	.0	.0	.0	.0
			3.0	-1.5	1.5	12.0	-6.0	.0	3.0	-1.5	-1.5							-18.0	-9.0	.0	.0	.0	.0
-1.5	.5	-.5	.0	.0	.0	1.5	-.5	-.5				.0	.0	.0	.0	.0	-2.0						
-1.5	.5	-.5	-6.0	2.0	.0	-1.5	.5	.5				.0	.0	.0	9.0	6.0	.0						
3.0	-1.5	1.5	12.0	-6.0	.0	3.0	-1.5	-1.5				.0	.0	.0	-18.0	-9.0	.0						
.0	.0	.0	1.5	-.5	-.5	.0	.0	.0		0		-1.5	-1.0	-1.0	.0	.0	.0			0			
-3.0	1.0	.0	-1.5	.5	.5	.0	.0	.0				4.5	3.0	1.0	.0	.0	.0						
6.0	-3.0	.0	3.0	-1.5	-1.5	.0	.0	.0				-9.0	-4.5	-1.5	.0	.0	.0						

Table 4

Latent roots	λ_1	λ_2	λ_3	λ_4	λ_5
	−10·5165	10·4967	8·7510	−8·7190	−5·7727
Latent vectors	x_1	x_2	x_3	x_4	x_5
	−0·0647	0·0150	0·0087	−0·0284	−0·2689
	0·0907	−0·0028	−0·0417	0·0082	0·3641
	0·1925	0·0217	−0·0823	−0·0043	0·4093
	−0·2143	0·1766	0·2040	−0·0771	−0·2136
	0·5833	0·2413	−0·5216	−0·2090	−0·3234
	0·9320	0·6792	−0·7714	−0·5946	−0·7574
	−0·2167	0·2256	−0·1624	0·0144	0·2152
	0·5524	0·3634	0·6026	0·0748	0·0366
	0·8401	0·9792	1·0000	0·2506	0·1824
	−0·0120	0·1134	−0·0307	0·1195	0·0060
	0·0284	0·1461	−0·0063	0·1392	0·5153
	0·0162	0·3774	−0·0552	0·3603	0·9882
	0·0147	−0·0661	−0·0306	0·0090	0·3292
	0·0045	−0·0933	−0·0125	0·0438	0·4179
	0·0167	0·2011	0·0051	−0·0880	−0·5305
	0·1800	−0·2186	−0·0825	0·2021	−0·0017
	−0·1751	−0·6302	0·2041	0·5401	−0·1817
	0·6071	1·0000	−0·5258	−0·8273	0·4606
	0·2082	−0·2256	0·0016	−0·1792	0·0567
	−0·3889	−0·5743	−0·0838	−0·6072	0·0525
	1·0000	0·8597	0·2629	1·0000	−0·1076
	0·1067	−0·0193	0·1149	−0·0363	0·0279
	−0·1389	−0·0221	−0·1360	0·0125	−0·5466
	0·3570	−0·0033	0·3488	−0·0726	1·0000

Table 2

-0.0032	-0.0067	-0.0071	0.2912	-0.2008	-0.0779	0.3456	-0.2157	-0.0741	-1.4313	0.9678	0.9776	-0.1291	-0.0498	-0.0390	-0.3302	-0.1215	-0.0783	1.2619	0.4143	0.4402	-0.0089	0.0003	-0.0046
0.0045	0.0094	0.0100	-0.4086	0.2817	0.1092	-0.4904	0.3025	0.1040	4.4037	-2.9549	-0.9685	0.1811	0.0699	0.0547	0.4631	0.1704	0.1097	-1.1660	-0.3797	-0.4162	-2.9875	-1.0005	0.0065
0.0096	0.0200	0.0213	-0.8668	0.5977	0.2318	-1.0405	0.6418	0.2205	8.7957	-4.4043	-1.4332	0.3843	0.1483	0.1160	0.9825	0.3616	0.2331	-2.2913	-1.2449	-1.3221	-5.9733	-3.0096	0.1382
-0.0107	-0.0223	-0.0237	0.9649	-0.6654	-0.2581	1.1584	-0.7145	1.7545	0.2275	-0.1065	-0.0744	-0.4278	-0.1651	-0.1292	0.4062	0.0097	0.2405	-0.7869	-0.2840	-0.1980	-1.5296	-0.4969	0.4846
0.0291	0.0607	0.0644	-2.6261	1.8108	0.7024	5.8476	-4.0556	0.6682	-0.6191	0.2898	0.2024	1.1643	0.4493	0.3515	1.4767	0.5956	0.2062	-3.8531	-1.2271	0.5389	-1.4194	-0.5029	0.5419
0.0465	0.0969	0.1029	-4.1962	2.8934	1.1222	12.9629	-5.8932	1.0677	-0.9892	0.4631	0.3233	1.8604	0.7180	0.5617	1.7564	0.2506	-0.3716	-8.5696	-4.7849	0.8611	-2.8712	-1.5046	1.5669
-0.0108	-0.0225	-0.0239	0.9756	-0.6727	1.7391	1.1711	-0.7223	-0.2482	0.2300	-0.1077	-0.0752	1.0675	0.3331	0.3693	-1.1059	-0.4070	-0.2624	-2.2976	-0.7872	0.2998	-0.0299	0.0011	-0.0156
0.0276	0.0575	0.0609	6.5130	-4.2851	0.6651	-2.9854	1.8414	0.6328	-0.5863	0.2745	0.1916	-0.3974	-0.0745	-0.1671	-3.1809	-0.9627	0.6688	0.5331	0.2320	1.0104	0.0764	-0.0028	0.0397
0.0419	0.0874	0.0927	14.2178	-6.3920	1.0115	-4.5402	2.8003	0.9624	-0.8916	0.4174	0.2914	-1.3232	-0.8529	-0.9937	-7.7129	-4.4221	1.0171	0.0919	-0.3868	2.2762	0.1161	-0.0042	0.0603
1.4994	-1.0012	0.9987	0.0539	-0.0371	-0.0144	0.0647	-0.0399	-0.0137	0.0127	-0.0059	-0.0042	-0.0239	-0.0092	-0.0072	-1.5611	-0.5225	0.4855	-0.0440	-0.0159	-0.0111	-0.0017	0.0001	-0.0009
4.5014	-2.9970	1.0031	-0.1277	0.0881	0.0342	-0.1534	0.0946	0.0325	-0.0301	0.0141	0.0090	-2.9434	-0.9781	0.0171	-1.3552	-0.4467	0.5344	0.1044	0.0376	0.0262	0.0039	-0.0001	0.0020
9.0001	-4.4983	1.5018	-0.0729	0.0503	0.0195	-0.0876	0.0540	0.0186	-0.0172	0.0081	0.0056	-5.9677	-2.9875	0.0098	-2.9173	-1.4697	1.5196	0.0060	0.0021	0.0015	0.0022	-0.0001	0.0012
0.0007	0.0015	0.0016	-0.0662	0.0456	0.0177	-1.5794	0.5490	-0.4832	-0.0156	0.0073	0.0051	0.0293	0.0113	0.0089	0.0750	0.0276	0.0178	0.0541	0.0195	0.0136	1.5020	0.9999	-0.9989
0.0002	0.0005	0.0005	-0.0205	0.0141	0.0055	-1.5246	0.5152	-0.4948	-3.0048	1.0023	0.0016	0.0091	0.0035	0.0027	0.0232	0.0085	0.0055	0.0167	0.0060	0.0042	4.5006	2.9999	-0.9997
-0.0008	0.0017	0.0018	-0.0752	0.0519	0.0201	2.9097	-1.4443	1.5191	5.9823	-2.9917	0.0058	0.0333	0.0127	0.0101	0.0852	0.0314	0.0202	0.0615	0.0221	0.0154	-8.9977	-4.5001	1.5012
0.0090	0.0187	0.0199	-2.3105	1.0588	-0.2832	-0.9729	0.6001	0.2062	1.3089	-0.4106	-0.4376	0.3593	0.1387	0.1085	0.9187	0.3381	0.2179	0.6626	0.2385	-1.8337	0.0249	-0.0009	0.0129
-0.0087	-0.0182	-0.0193	-0.7118	-0.5435	-0.7108	-5.0539	1.4164	-0.2005	-1.3142	0.4130	0.4393	-0.3494	-0.1349	-0.1055	-0.8934	-0.3288	-0.2120	0.3557	5.7680	-0.1617	-0.0242	0.0009	-0.0127
0.0303	0.0631	0.0670	0.2666	0.385	2.2310	8.7188	-3.9762	0.6955	2.3556	-1.1983	-1.2894	1.2119	0.4677	0.3660	3.0983	1.1403	0.7350	-15.7654	-8.1955	0.5609	0.0839	-0.0030	0.0436
-1.4896	0.5217	-0.4770	-0.9373	0.6463	0.2507	0.3749	0.1940	-0.2615	-0.2210	0.1034	0.0722	0.4156	0.1604	0.1254	1.0624	0.3910	-1.7479	0.7662	0.2759	0.1923	0.0289	-0.0010	0.0149
-1.5194	0.4596	-0.5429	-4.2491	0.7927	-0.4683	0.6018	-0.7963	0.0545	0.4127	-0.1932	-0.1349	-0.7763	-0.2996	-0.2344	7.0154	5.3596	-0.4708	-1.4313	-0.5153	-0.3593	-0.0536	0.0019	-0.0279
3.0499	-1.3960	1.6104	7.4977	-2.8955	1.2041	-2.4046	1.8335	-0.3544	-1.0614	0.4969	0.3469	1.9961	0.7703	0.6027	-12.8966	-7.1217	1.2107	3.6806	1.3251	0.9239	0.1382	-0.0050	0.0718
0.0053	0.0111	0.0119	1.0195	-0.1687	-0.3715	-0.5768	0.3558	0.1223	-0.1133	0.0530	0.0370	-1.2870	-0.9178	-0.9357	0.5449	0.2005	0.1292	0.3928	0.1414	0.0986	0.0148	-0.0005	0.0077
-3.0069	0.9856	-0.0153	-0.8748	0.0689	0.3328	0.7505	-0.4629	-0.1591	0.1474	-0.0690	-0.0482	4.2228	2.8930	0.9163	-0.7087	-0.2608	-0.1681	-0.5111	-0.1840	-0.1283	-0.0191	0.0007	-0.0099
6.0178	-2.9629	0.0394	1.3928	-0.3918	-1.0702	-1.9293	1.1899	0.4089	-0.3789	0.1774	0.1238	-8.2874	-4.2250	-1.2849	1.8218	0.6705	0.4322	1.3139	0.4730	0.3298	0.0493	-0.0018	0.0256

Table 3

-0.0326	0.0046	-0.0160	0.2203	-0.1775	-0.0959	0.2933	-0.1542	-0.0670	-1.4332	0.9678	0.9764	-0.1306	-0.0486	-0.0409	-0.2655	-0.07804	-0.0957	1.3416	0.4627	0.4237	0.0068	0.0077	-0.0097
0.0100	0.0073	0.0117	-0.3952	0.2773	0.1127	-0.4798	0.2965	0.1064	4.4040	-2.9549	-0.9683	0.1814	0.6967	0.0551	0.4509	0.1623	0.1131	-1.1810	-0.3809	-0.4130	-2.9904	-1.0018	0.0075
-0.0326	0.0364	0.0085	-0.9693	0.6314	0.2058	-1.1219	0.6728	0.2019	8.7929	-4.4045	-1.4349	0.3821	0.1501	0.1132	1.0756	0.4244	0.2080	-2.1763	-1.1749	-1.3459	-5.9506	-2.9904	0.0065
-0.3554	0.1107	-0.1273	0.1313	-0.3910	-0.4701	0.4960	-0.4623	1.6027	0.2049	-0.1077	-0.0877	-0.4453	-0.1503	-0.1519	1.1653	0.6061	0.0365	0.1477	0.2650	-0.3918	-1.3446	-0.4128	0.4247
-0.4420	0.2424	-0.0772	-3.7656	2.1859	0.4126	4.9422	-3.7110	0.4608	-0.6499	0.2882	0.1841	1.1405	0.4695	0.3205	2.5142	1.2936	-0.0726	-2.5731	-0.4493	0.2740	-1.1665	-0.3851	0.4600
-1.2793	0.6085	-0.2956	-7.4030	3.9488	0.3068	10.4159	-4.4923	0.4838	-1.0760	0.4504	0.2719	1.7933	0.7747	0.4743	4.6764	2.2151	-1.1553	-4.9669	-2.5761	0.1156	-2.1594	-1.1732	1.3364
-0.4511	0.1474	-0.1563	-0.0895	-0.3222	1.4362	0.3249	-0.4002	-0.4421	0.2012	-0.1092	-0.0922	1.0452	0.3519	0.3404	-0.1361	0.2455	-0.5230	-1.1010	-0.0602	0.0522	0.2065	0.1112	-0.0921
-0.6817	0.3311	-0.1522	4.7974	-3.7205	0.2289	-4.3486	2.3603	0.3204	-0.6327	0.2720	0.1641	-0.4333	-0.0441	-0.2139	-1.6188	0.0886	0.2490	2.4605	1.4030	0.6115	0.4572	-0.1746	-0.0837
-1.8690	0.8247	-0.4816	9.5955	-4.6707	-0.1639	-8.2128	4.1984	0.1208	-1.0168	0.4107	0.2172	-1.4198	-0.7710	-1.1197	-3.5040	-1.5905	-0.1139	5.2848	2.7681	1.2015	1.1421	0.4736	-0.2720
1.2781	-0.9159	0.9322	-0.4814	0.1390	-0.1505	-0.3607	0.1220	-0.1112	-0.0018	-0.0067	-0.0127	-0.0351	0.0003	-0.0218	-1.0736	-0.1945	0.3545	0.5573	0.3495	-0.1355	0.1172	0.0554	-0.0393
4.2162	-2.8870	0.9174	-0.8175	0.3151	-0.1412	-0.7014	0.3032	-0.0931	-0.0488	0.0131	-0.0012	-2.9578	-0.9659	-0.0017	-0.7271	-0.0241	0.3656	0.8794	0.5084	-0.1341	0.1570	0.0712	-0.0475
8.2642	-4.2141	1.2804	-1.8548	0.6367	-0.4336	-1.5033	0.5929	-0.3058	-0.0654	0.0055	-0.0230	-6.6005	-2.9560	-0.0388	-1.2949	-0.3780	1.0836	2.0614	1.2376	-0.3993	0.3977	0.1841	-0.1269
0.1298	-0.0483	0.0404	0.2460	-0.0571	0.0971	-1.3314	0.4546	-0.4263	-0.0072	0.0078	0.0101	0.0359	0.0058	0.0174	-0.2093	-0.1636	0.0942	-0.2966	-0.1936	0.0862	1.4327	0.9677	-0.9765
0.1844	-0.0706	0.0559	0.4249	-0.1325	0.1187	-1.1707	0.3804	-0.4137	-2.9928	1.0029	0.0087	0.0184	-0.0044	0.0149	-0.3824	-0.2643	0.1145	-0.4837	-0.2980	0.1076	4.4018	2.9539	-0.9677
-0.3917	0.1532	-0.1161	-1.0246	0.3643	-0.2213	2.1554	-1.1572	1.3463	5.9566	-2.9931	-0.0094	0.0135	0.0297	-0.0158	0.9497	0.6130	-0.2121	1.1281	0.6702	-0.2053	-8.7870	-4.4019	1.4329
0.4357	-0.1459	0.1481	-1.2763	0.7191	-0.0208	-0.1528	0.2879	0.3941	1.3369	-0.4091	-0.4210	0.3609	0.1204	0.1366	-0.0211	-0.2942	0.4705	-0.4970	-0.4659	-1.5937	-0.2042	-0.1076	0.0871
1.2213	-0.4928	0.3504	2.2636	-1.5227	0.0458	-2.6898	0.5165	0.3412	-1.2336	0.4173	0.4670	-0.2672	-0.1875	-0.0244	-3.6026	-2.1516	0.5161	5.0130	3.7372	0.5300	-0.6846	-0.3067	0.2013
-1.9215	0.8163	-0.5196	-4.4546	1.9386	1.0305	4.9676	-2.5483	-0.1640	2.2278	-1.2052	-1.3652	1.1131	0.5513	0.2372	7.3972	4.0326	-0.4201	-10.4615	-4.5731	-0.5367	1.1319	0.4850	-0.2958
-1.0494	0.3518	-0.3447	0.1276	0.2958	0.5215	1.2210	-0.1281	-0.0676	-0.1921	0.1050	0.0893	0.4378	0.1415	0.1545	0.0928	-0.2614	-1.4874	-0.4301	-0.4510	0.4399	-0.2076	-0.1111	0.0915
-0.3986	0.0271	-0.2060	-1.5380	-0.0996	0.2211	2.7559	-1.6163	0.5481	0.4862	-0.1893	-0.0914	-0.7196	-0.3476	-0.1605	4.5467	3.6087	0.1925	-4.4771	-2.3658	0.2710	-0.6555	-0.2783	0.1670
1.3721	-0.7486	1.1061	3.4393	-1.5598	0.1721	-5.6292	3.0610	-1.0933	-1.1713	0.4910	0.2816	1.9113	0.8422	0.4921	-9.2012	-4.6355	0.2177	8.2399	4.0952	-0.0196	1.0391	0.4145	-0.2199
0.0431	-0.0035	0.0231	1.1108	-0.1987	-0.3483	-0.5043	0.3282	0.1389	-0.1108	0.0532	0.0385	-1.2850	-0.9194	-0.9332	0.4616	0.1445	0.1516	0.2903	0.0791	0.1198	-0.0055	-0.0099	0.0142
-2.9636	0.9689	-0.0024	-0.7704	0.0345	0.3593	0.8335	-0.4945	-0.1401	0.1502	-0.0689	-0.0465	4.2250	2.8912	0.9191	-0.8037	-0.3248	-0.1426	-0.6284	-0.2552	-0.1041	-0.0424	-0.0101	-0.0025
6.0242	-2.9653	0.0413	1.4083	-0.3969	-1.0662	-1.9170	0.1185	0.4118	-0.3785	0.1774	0.1241	-8.2871	-4.2253	-1.2844	1.8077	0.6610	0.4360	1.2965	0.4625	0.3334	0.0459	-0.0034	0.0267

given below are summarized in **Table 4**. All computations were performed in floating point with 8 significant figures* on the Olivetti ELEA 6001 digital computer. Iterations stopped when the absolute difference $|\Delta\lambda^{(i)}|$ between the normalization factors $\lambda^{(i-1)}$, $\lambda^{(i)}$ of two successive iterates became less than the order of magnitude of $10^{-7}|\lambda^{(i)}|$.

Example 1

The first dominant latent roots

$$\lambda_1 = -10 \cdot 5165$$
$$\lambda_2 = \quad 10 \cdot 4967$$

of the given matrix (Table 1) are very close in absolute magnitude:

$$\left|\frac{\lambda_2}{\lambda_1}\right| = 0 \cdot 9981.$$

Iteration without acceleration therefore converges very slowly (e.g. after 100 premultiplications of the starting vector (1, 1, . . ., 1) by the given matrix, the difference $|\Delta\lambda^{(i)}|$ between two successive normalization factors decreased from $11 \cdot 21$ to $9 \cdot 38$. We recall that the output $|\Delta\lambda^{(i)}|$ must be less in magnitude than 10^{-6}, in the present case).

After 7 initial iterations the accelerating procedure gives the following values:

$$\lambda_1 \approx -11 \cdot 8821$$
$$\lambda_2 \approx \quad 11 \cdot 8729$$
$$|r| = \left|\frac{\lambda_2}{\lambda_1}\right| \approx 0 \cdot 9992$$

and taking $m = 2$, formula (7) determines the value

$$q = 3 \cdot 9586$$

for the accelerating parameter.

The new rate of convergence becomes

$$|r| = 0 \cdot 8758 \tag{12}$$

and the output is obtained after 51 accelerated iterations. This accelerated process does not give high-speed convergence; nevertheless, it should be noted that a considerably better choice for q is limited by the subdominant roots

$$\lambda_3 = 8 \cdot 7510$$
$$\lambda_4 = -8 \cdot 7190.$$

It is easy to see—a posteriori—that the mostly reduced rates of convergence are

$$|r| = 0 \cdot 8466 \tag{13}$$
$$|r| = 0 \cdot 8424 \tag{14}$$

which differs but slightly from (12) given by this accelerating technique.

* For the sake of a simpler representation, the reduced matrices and most of the results are given with four decimals.

Moreover, it will be seen below in the comments on Example 2 that the convergence rate (13) obtained for $|\lambda_1 - q| = |\lambda_3 - q|$, and that (14) obtained for $|\lambda_2 - q| = |\lambda_4 - q|$, do not assure the minimum number of iterations.

Example 2

For the given matrix (Table 2) the first two latent roots have the following values:

$$\begin{aligned} \lambda_1 &= 10 \cdot 4967 \\ \lambda_2 &= \ \ 8 \cdot 7510. \end{aligned} \tag{15}$$

Iteration without acceleration gives a convergence rate

$$|r| = \left|\frac{\lambda_2}{\lambda_1}\right| = 0 \cdot 8337 \tag{16}$$

and needs 78 repeated matrix-by-vector multiplications starting with the vector (1, 1, . . ., 1).

The third root being

$$\lambda_3 = -8 \cdot 7190 \tag{17}$$

it is easy to verify that in practice there is no way to improve the rate of convergence by a transformation $A - qI$. Our technique, applied after the first 7 iterations, gives for the first two latent roots the values

$$\lambda_1 \approx 10 \cdot 3581$$
$$\lambda_2 \approx -8 \cdot 8791 \quad \text{(this refers to } \lambda_3\text{)},$$

gives the rate

$$|r| = 0 \cdot 8572,$$

determines

$$q = -3 \cdot 1076 \quad (\text{for } m = 2), \tag{18}$$

and converges after 38 accelerated iterations.

The value of q (18) makes the convergence rate worse, as the new rate

$$\left|\frac{\lambda_2 - q}{\lambda_1 - q}\right| = 0 \cdot 8717$$

exceeds the initial one (16).*

In spite of the above-mentioned worsening of the convergence rate, acceleration was obtained by reducing the total number of iterations from 78 to 45.

Table 5 illustrates partial comparative tracings of the iterative process both without and with acceleration.

One may justify this fact by examining the spectral expansion of the current iterate y_k as it is done in the formula (1′). In fact, by (15), (17)

$$\left|\frac{\lambda_2}{\lambda_1}\right| \approx \left|\frac{\lambda_3}{\lambda_1}\right|$$

and it follows that there are two vectors x_2, x_3 which have non-negligible contributions in y_k's expansion during the whole process. This retards convergence to

* This important fact was pointed out by the referee.

the dominant latent vector x_1.*

In the accelerated process, the shifted latent roots

$$\lambda_1' = \lambda_1 - q = 13\cdot6043$$
$$\lambda_2' = \lambda_2 - q = 11\cdot8586$$
$$\lambda_3' = \lambda_3 - q = -5\cdot6114$$

present but one dangerous vector x_2 to be removed.

In closing these comments we may conclude that the accelerating procedure takes into consideration the distribution of more than two of the appropriate latent roots when applied after a small number of initial iterations when the convergence is not yet geometric. This is due to the fact that the shift q is deduced from the outline of the convergence determined by all factors which play a greater or a smaller role in the considered matrix power process.

Example 3

The matrix is given in Table 3. As the first two latent roots are

$$\lambda_1 = 8\cdot7510$$
$$\lambda_2 = -8\cdot7190,$$

iteration without acceleration gives the convergence rate

$$|r| = \left|\frac{\lambda_2}{\lambda_1}\right| = 0\cdot9963$$

which assures, as in Example 1, a very slow convergence.

The acceleration applied after 11 premultiplications of the starting vector (1, 1, . . ., 1) by the given matrix supplies the estimate

$$|r| = 0\cdot9837$$

and gives

$$q = 2\cdot8728.$$

* A similar case may occur in Example 1, for the best rates (13), (14).

Table 5

ITERATION NUMBER i	$\lvert\Delta\lambda^{(i)}\rvert$		REMARKS
	WITHOUT ACCELERATION	WITH ACCELERATION	
10	0·974012	0·364757	
20	0·356286	0·004830	
30	0·055547	0·000086	
44	0·004136	0·000013*	* followed by output
50	0·001361	—	
60	0·000214	—	
70	0·000033	—	
77	0·000010**	—	** followed by output

Having $\lambda_3 = -5\cdot7727$, the new convergence rate becomes

$$|r| = \left|\frac{\lambda_3 - q}{\lambda_2 - q}\right| = 0\cdot7458,$$

and convergence to the negative root λ_2 is obtained after 18 accelerated iterations.

Acknowledgement

This work has been carried out as part of the research programme of Olivetti-General Electric in Milan. The author would like to thank his colleague Dr. M. Menconi whose suggestions have greatly improved the paper. The author is also grateful to the referee for useful criticism and suggestions concerning the paper.

References

Bodewig, E. (1959). *Matrix Calculus* (second edition), Amsterdam: North-Holland Publishing Company.

Fox, L. (1964). *An Introduction to Numerical Linear Algebra*, Oxford: Clarendon Press.

Osborne, E. E. (1958). "On Acceleration and Matrix Deflation Processes Used with the Power Method," *J. Soc. Indust. Appl. Math.*, Vol. 6, pp. 279–87.

Wilkinson, J. H. (1954). "The Calculation of the Latent Roots and Vectors of Matrices on the Pilot Model of the ACE," *Proc. Camb. Phil. Soc.*, Vol. 50, pp. 536–66.