

ESTUDIO DE LOS ACIDOS 3-NAFTIL-2-MERCAPROPENOICOS
COMO COMPLEJANTES DE IONES METALICOS

I. 3-(1-NAFTIL)-2-MERCAPTOPROPENOICO (31N2MP)

II. 3-(2-NAFTIL)-2-MERCAPTOPROPENOICO (32N2MP)

Memoria presentada por
JOSE LUIS BELTRAN ABADIA
para aspirar al grado de
Doctor en Química

Barcelona, octubre de 1986

ESTUDIO DE LOS ACIDOS 3-NAFTIL-2-MERCAPROPENOICOS
COMO COMPLEJANTES DE IONES METALICOS

I. 3-(1-NAFTIL)-2-MERCAPTOPROPENOICO (31N2MP)

II. 3-(2-NAFTIL)-2-MERCAPTOPROPENOICO (32N2MP)



Memoria presentada por
JOSE LUIS BELTRAN ABADIA
para aspirar al grado de
Doctor en Química

Barcelona, octubre de 1986

Serie ES.NI1N.1

3.216E-5	0.0	7.041E-5	0.0	3.274	1.0		
0.388	0.393	0.484	0.561	0.602	0.581	0.488	0.350
0.211	0.099	0.044	0.023	0.015	0.011	0.007	0.005
0.005	0.004	0.003	0.003	0.002	0.002	0.002	0.001
3.216E-5	0.0	7.041E-5	0.0	3.839	1.0		
0.403	0.459	0.552	0.612	0.636	0.598	0.489	0.339
0.202	0.100	0.062	0.052	0.048	0.046	0.042	0.039
0.035	0.030	0.025	0.020	0.015	0.011	0.008	0.006
3.216E-5	0.0	7.041E-5	0.0	4.106	1.0		
0.442	0.510	0.595	0.641	0.657	0.614	0.506	0.355
0.219	0.122	0.090	0.090	0.090	0.091	0.089	0.081
0.073	0.062	0.051	0.040	0.028	0.019	0.013	0.009
3.216E-5	0.0	7.041E-5	0.0	4.272	1.0		
0.497	0.581	0.659	0.690	0.698	0.650	0.539	0.385
0.250	0.156	0.132	0.141	0.147	0.149	0.145	0.135
0.119	0.102	0.084	0.065	0.046	0.031	0.021	0.015
3.216E-5	0.0	7.041E-5	0.0	4.411	1.0		
0.530	0.624	0.695	0.716	0.717	0.668	0.556	0.402
0.269	0.179	0.158	0.170	0.178	0.183	0.181	0.169
0.151	0.128	0.105	0.082	0.057	0.038	0.025	0.017
3.216E-5	0.0	7.041E-5	0.0	4.696	1.0		
0.606	0.709	0.771	0.781	0.769	0.712	0.591	0.440
0.314	0.231	0.218	0.233	0.247	0.252	0.248	0.232
0.208	0.178	0.146	0.113	0.079	0.052	0.035	0.024
3.216E-5	0.0	7.041E-5	0.0	4.977	1.0		
0.672	0.806	0.850	0.831	0.810	0.753	0.641	0.484
0.356	0.275	0.273	0.299	0.324	0.337	0.332	0.312
0.278	0.237	0.194	0.147	0.101	0.065	0.040	0.026

Serie ES.NI1N.2

3.216E-5	0.0	9.261E-5	0.0	3.229	1.0		
0.470	0.524	0.648	0.745	0.800	0.777	0.656	0.465
0.275	0.123	0.050	0.024	0.016	0.013	0.011	0.009
0.008	0.007	0.007	0.006	0.005	0.004	0.003	0.003
3.216E-5	0.0	9.261E-5	0.0	3.689	1.0		
0.517	0.585	0.708	0.792	0.827	0.781	0.640	0.445
0.265	0.130	0.076	0.060	0.055	0.049	0.044	0.039
0.034	0.029	0.024	0.019	0.013	0.009	0.006	0.004
3.216E-5	0.0	9.261E-5	0.0	4.246	1.0		
0.646	0.758	0.858	0.897	0.905	0.846	0.700	0.497
0.322	0.199	0.170	0.175	0.186	0.190	0.184	0.172
0.153	0.130	0.105	0.080	0.054	0.034	0.020	0.012
3.216E-5	0.0	9.261E-5	0.0	4.549	1.0		
0.740	0.878	0.962	0.974	0.968	0.902	0.755	0.550
0.377	0.261	0.244	0.261	0.280	0.289	0.284	0.264
0.235	0.200	0.162	0.123	0.085	0.052	0.031	0.019
3.216E-5	0.0	9.261E-5	0.0	4.937	1.0		
0.839	1.004	1.072	1.060	1.038	0.964	0.814	0.607
0.436	0.328	0.320	0.350	0.379	0.394	0.389	0.365
0.326	0.276	0.224	0.170	0.117	0.072	0.042	0.025
3.216E-5	0.0	9.261E-5	0.0	5.305	1.0		
0.877	1.042	1.106	1.095	1.065	0.986	0.827	0.626
0.461	0.359	0.352	0.382	0.411	0.423	0.416	0.390
0.347	0.295	0.239	0.181	0.125	0.077	0.046	0.028
3.216E-5	0.0	9.261E-5	0.0	6.573	1.0		
0.941	1.125	1.174	1.139	1.103	1.025	0.872	0.668
0.502	0.400	0.399	0.437	0.474	0.491	0.485	0.455
0.405	0.345	0.279	0.211	0.146	0.091	0.055	0.034
3.216E-5	0.0	9.261E-5	0.0	7.334	1.0		
0.947	1.129	1.172	1.131	1.096	1.024	0.879	0.677
0.512	0.410	0.408	0.446	0.481	0.498	0.491	0.460
0.410	0.349	0.283	0.214	0.148	0.093	0.056	0.035

Serie ES.NI1N.3

8.040E-6	0.0	7.285E-5	0.0	6.094	1.0			
0.470	0.553	0.633	0.667	0.673	0.626	0.510	0.357	
0.225	0.135	0.114	0.118	0.126	0.128	0.126	0.117	
0.104	0.089	0.072	0.054	0.037	0.023	0.013	0.007	
1.608E-5	0.0	6.475E-5	0.0	6.106	1.0			
0.553	0.658	0.706	0.704	0.690	0.639	0.537	0.399	
0.286	0.213	0.206	0.223	0.242	0.251	0.248	0.231	
0.206	0.175	0.142	0.107	0.075	0.046	0.028	0.018	
2.412E-5	0.0	5.666E-5	0.0	6.115	1.0			
0.596	0.722	0.743	0.708	0.678	0.631	0.542	0.419	
0.322	0.264	0.271	0.301	0.329	0.343	0.340	0.319	
0.282	0.239	0.193	0.144	0.098	0.059	0.033	0.019	
3.216E-5	0.0	4.856E-5	0.0	6.104	1.0			
0.539	0.654	0.665	0.624	0.594	0.553	0.481	0.378	
0.298	0.252	0.261	0.289	0.314	0.326	0.321	0.301	
0.266	0.225	0.182	0.136	0.093	0.055	0.032	0.018	
4.020E-5	0.0	4.047E-5	0.0	6.085	1.0			
0.457	0.553	0.561	0.526	0.502	0.469	0.408	0.322	
0.256	0.218	0.225	0.247	0.267	0.276	0.270	0.252	
0.223	0.189	0.153	0.114	0.079	0.047	0.027	0.016	
4.824E-5	0.0	3.238E-5	0.0	6.106	1.0			
0.362	0.438	0.444	0.414	0.393	0.368	0.322	0.257	
0.205	0.177	0.183	0.199	0.215	0.220	0.214	0.200	
0.177	0.150	0.121	0.091	0.063	0.038	0.023	0.014	
5.628E-5	0.0	2.428E-5	0.0	6.053	1.0			
0.263	0.317	0.320	0.297	0.283	0.266	0.234	0.189	
0.152	0.133	0.136	0.147	0.156	0.159	0.153	0.142	
0.126	0.106	0.086	0.064	0.043	0.027	0.016	0.010	
6.432E-5	0.0	1.619E-5	0.0	6.143	1.0			
0.185	0.222	0.224	0.208	0.197	0.185	0.165	0.133	
0.109	0.095	0.098	0.104	0.110	0.110	0.107	0.100	
0.090	0.076	0.061	0.045	0.031	0.019	0.012	0.008	
7.236E-5	0.0	8.904E-6	0.0	6.076	1.0			
0.084	0.101	0.103	0.095	0.090	0.086	0.079	0.064	
0.054	0.048	0.048	0.051	0.053	0.050	0.048	0.044	
0.039	0.033	0.027	0.020	0.014	0.009	0.006	0.004	

Serie ES.NIIN.4

4.020E-5	0.0	1.619E-5	0.0	6.094	1.0		
0.176	0.210	0.213	0.200	0.192	0.177	0.156	0.124
0.100	0.087	0.088	0.095	0.102	0.101	0.099	0.091
0.080	0.068	0.055	0.041	0.028	0.016	0.011	0.007
4.020E-5	0.0	3.238E-5	0.0	6.127	1.0		
0.365	0.438	0.444	0.417	0.398	0.366	0.319	0.254
0.202	0.173	0.180	0.195	0.211	0.216	0.211	0.196
0.173	0.147	0.119	0.089	0.060	0.035	0.021	0.013
4.020E-5	0.0	4.856E-5	0.0	6.065	1.0		
0.550	0.662	0.669	0.629	0.600	0.553	0.481	0.381
0.303	0.259	0.268	0.293	0.316	0.325	0.319	0.297
0.263	0.222	0.180	0.134	0.091	0.054	0.031	0.018
4.020E-5	0.0	6.475E-5	0.0	6.106	1.0		
0.717	0.868	0.880	0.828	0.790	0.730	0.632	0.496
0.391	0.331	0.343	0.380	0.413	0.430	0.424	0.397
0.353	0.298	0.241	0.180	0.123	0.073	0.042	0.024
4.020E-5	0.0	8.094E-5	0.0	6.048	1.0		
0.884	1.072	1.094	1.032	0.987	0.911	0.786	0.612
0.476	0.399	0.413	0.460	0.504	0.525	0.521	0.488
0.434	0.369	0.297	0.223	0.152	0.091	0.051	0.030
4.020E-5	0.0	9.713E-5	0.0	6.080	1.0		
1.046	1.265	1.302	1.241	1.192	1.099	0.943	0.728
0.556	0.458	0.470	0.522	0.573	0.599	0.593	0.557
0.497	0.423	0.343	0.257	0.175	0.106	0.060	0.035
4.020E-5	0.0	1.133E-4	0.0	6.040	1.0		
1.127	1.361	1.419	1.372	1.326	1.225	1.043	0.795
0.594	0.474	0.481	0.533	0.584	0.612	0.606	0.570
0.509	0.434	0.352	0.266	0.181	0.110	0.063	0.037
4.020E-5	0.0	1.295E-4	0.0	5.960	1.0		
1.227	1.468	1.554	1.526	1.483	1.368	1.153	0.870
0.641	0.498	0.495	0.544	0.593	0.618	0.612	0.576
0.514	0.439	0.357	0.270	0.185	0.112	0.065	0.038
4.020E-5	0.0	1.619E-4	0.0	5.996	1.0		
1.387	1.659	1.793	1.793	1.764	1.629	1.366	1.008
0.708	0.515	0.498	0.545	0.597	0.626	0.623	0.587
0.526	0.449	0.367	0.278	0.191	0.117	0.067	0.040

Anexo 5.2: datos espectroscópicos del sistema Co(II)-31N2MP

Serie ES.CO1N.1

1.808E-5	0.0	9.514E-5	0.0	4.195	1.0		
0.496	0.568	0.704	0.795	0.828	0.773	0.623	0.418
0.233	0.096	0.045	0.031	0.027	0.021	0.018	0.016
0.015	0.012	0.010	0.008	0.006	0.005	0.005	0.004
1.808E-5	0.0	9.514E-5	0.0	5.128	1.0		
0.563	0.651	0.773	0.841	0.850	0.791	0.663	0.499
0.348	0.236	0.186	0.164	0.153	0.144	0.132	0.118
0.100	0.080	0.062	0.046	0.035	0.027	0.022	0.018
1.808E-5	0.0	9.514E-5	0.0	5.542	1.0		
0.622	0.723	0.842	0.899	0.900	0.843	0.726	0.573
0.431	0.327	0.280	0.261	0.251	0.242	0.228	0.205
0.175	0.139	0.107	0.080	0.061	0.046	0.037	0.033
1.808E-5	0.0	9.514E-5	0.0	5.963	1.0		
0.696	0.814	0.929	0.976	0.970	0.910	0.801	0.659
0.524	0.426	0.384	0.372	0.368	0.363	0.347	0.315
0.268	0.215	0.166	0.126	0.096	0.073	0.058	0.050
1.808E-5	0.0	9.514E-5	0.0	6.283	1.0		
0.726	0.848	0.961	1.003	0.995	0.936	0.831	0.695
0.563	0.469	0.429	0.419	0.419	0.413	0.396	0.360
0.308	0.245	0.188	0.141	0.106	0.081	0.066	0.057
1.808E-5	0.0	9.514E-5	0.0	6.595	1.0		
0.730	0.855	0.964	1.002	0.991	0.935	0.832	0.700
0.573	0.480	0.443	0.435	0.437	0.433	0.415	0.378
0.322	0.255	0.194	0.143	0.106	0.080	0.063	0.055
1.808E-5	0.0	9.514E-5	0.0	6.846	1.0		
0.738	0.857	0.963	1.001	0.993	0.937	0.836	0.705
0.578	0.485	0.449	0.441	0.440	0.432	0.412	0.374
0.319	0.253	0.193	0.144	0.108	0.082	0.066	0.057
1.808E-5	0.0	9.514E-5	0.0	7.199	1.0		
0.740	0.856	0.958	0.996	0.988	0.935	0.836	0.707
0.583	0.491	0.455	0.445	0.443	0.434	0.413	0.374
0.318	0.252	0.192	0.143	0.107	0.080	0.065	0.056

Serie ES.C01N.2

7.232E-6	0.0	6.719E-5	0.0	6.623	1.0			
0.432	0.496	0.573	0.612	0.620	0.586	0.508	0.401	
0.303	0.228	0.198	0.183	0.173	0.162	0.147	0.132	
0.112	0.090	0.070	0.053	0.041	0.033	0.028	0.024	
1.085E-5	0.0	6.345E-5	0.0	6.650	1.0			
0.453	0.525	0.595	0.624	0.624	0.593	0.527	0.437	
0.353	0.290	0.262	0.250	0.243	0.233	0.218	0.196	
0.167	0.133	0.103	0.077	0.059	0.046	0.038	0.033	
1.446E-5	0.0	5.972E-5	0.0	6.636	1.0			
0.462	0.537	0.602	0.623	0.616	0.586	0.530	0.455	
0.383	0.329	0.304	0.297	0.293	0.286	0.271	0.244	
0.208	0.167	0.129	0.098	0.074	0.057	0.048	0.041	
1.808E-5	0.0	5.599E-5	0.0	6.650	1.0			
0.447	0.523	0.584	0.602	0.593	0.565	0.515	0.447	
0.381	0.332	0.310	0.304	0.303	0.298	0.284	0.256	
0.217	0.173	0.132	0.099	0.074	0.057	0.047	0.039	
2.170E-5	0.0	5.226E-5	0.0	6.652	1.0			
0.424	0.498	0.555	0.571	0.561	0.535	0.490	0.428	
0.367	0.321	0.301	0.298	0.298	0.294	0.281	0.253	
0.215	0.171	0.130	0.097	0.072	0.055	0.045	0.038	
2.893E-5	0.0	4.479E-5	0.0	6.648	1.0			
0.365	0.429	0.480	0.494	0.485	0.462	0.422	0.369	
0.316	0.276	0.259	0.256	0.257	0.255	0.245	0.221	
0.188	0.150	0.115	0.085	0.064	0.049	0.041	0.034	
3.616E-5	0.0	3.733E-5	0.0	6.623	1.0			
0.309	0.359	0.401	0.411	0.405	0.388	0.355	0.313	
0.271	0.237	0.223	0.217	0.215	0.210	0.199	0.181	
0.154	0.123	0.095	0.072	0.055	0.043	0.036	0.032	
4.339E-5	0.0	2.986E-5	0.0	6.601	1.0			
0.244	0.282	0.314	0.324	0.319	0.307	0.282	0.249	
0.215	0.190	0.178	0.170	0.167	0.162	0.152	0.137	
0.118	0.095	0.074	0.056	0.044	0.035	0.031	0.027	
5.062E-5	0.0	2.240E-5	0.0	6.639	1.0			
0.189	0.218	0.242	0.248	0.245	0.240	0.221	0.197	
0.171	0.152	0.143	0.138	0.135	0.130	0.123	0.111	
0.096	0.078	0.061	0.048	0.038	0.031	0.028	0.024	

Serie ES.CD1N.2 (continuación)

5.786E-5	0.0	1.493E-5	0.0	6.619	1.0		
0.126	0.144	0.159	0.163	0.162	0.159	0.148	0.131
0.116	0.103	0.096	0.090	0.086	0.079	0.074	0.067
0.058	0.048	0.039	0.032	0.026	0.022	0.021	0.019
6.509E-5	0.0	7.465E-6	0.0	6.587	1.0		
0.069	0.076	0.083	0.086	0.087	0.089	0.085	0.077
0.071	0.064	0.059	0.055	0.050	0.044	0.040	0.036
0.032	0.028	0.025	0.022	0.020	0.017	0.018	0.017

Serie ES.CO1N.3

3.616E-5	0.0	8.846E-6	0.0	6.569	1.0		
0.079	0.090	0.099	0.102	0.102	0.100	0.094	0.083
0.074	0.066	0.060	0.057	0.054	0.048	0.044	0.040
0.035	0.030	0.025	0.021	0.019	0.016	0.017	0.016
3.616E-5	0.0	1.769E-5	0.0	6.603	1.0		
0.152	0.172	0.188	0.191	0.188	0.182	0.169	0.149
0.130	0.114	0.105	0.100	0.096	0.090	0.084	0.076
0.066	0.054	0.043	0.034	0.027	0.023	0.022	0.019
3.616E-5	0.0	2.654E-5	0.0	6.596	1.0		
0.215	0.249	0.277	0.284	0.281	0.269	0.247	0.219
0.189	0.167	0.155	0.150	0.146	0.140	0.132	0.119
0.102	0.082	0.064	0.049	0.038	0.030	0.027	0.024
3.616E-5	0.0	3.539E-5	0.0	6.531	1.0		
0.285	0.331	0.367	0.376	0.370	0.354	0.324	0.287
0.249	0.219	0.205	0.199	0.195	0.189	0.178	0.161
0.137	0.109	0.084	0.064	0.049	0.038	0.033	0.029
3.616E-5	0.0	4.423E-5	0.0	6.584	1.0		
0.362	0.425	0.475	0.487	0.477	0.453	0.414	0.362
0.311	0.272	0.256	0.254	0.255	0.253	0.242	0.220
0.187	0.150	0.115	0.086	0.065	0.050	0.042	0.036
3.616E-5	0.0	5.308E-5	0.0	6.566	1.0		
0.435	0.512	0.569	0.582	0.570	0.543	0.498	0.437
0.377	0.333	0.313	0.311	0.312	0.308	0.296	0.268
0.228	0.182	0.139	0.104	0.079	0.060	0.051	0.043
3.616E-5	0.0	6.192E-5	0.0	6.542	1.0		
0.503	0.594	0.662	0.678	0.666	0.633	0.577	0.505
0.433	0.381	0.358	0.357	0.360	0.359	0.346	0.315
0.268	0.214	0.164	0.122	0.093	0.070	0.058	0.049
3.616E-5	0.0	7.077E-5	0.0	6.558	1.0		
0.592	0.696	0.777	0.796	0.780	0.741	0.678	0.595
0.511	0.450	0.424	0.424	0.428	0.426	0.411	0.374
0.318	0.254	0.194	0.144	0.108	0.082	0.067	0.058
3.616E-5	0.0	8.846E-5	0.0	6.531	1.0		
0.749	0.882	0.984	1.008	0.990	0.939	0.858	0.752
0.644	0.566	0.535	0.534	0.542	0.542	0.521	0.477
0.407	0.324	0.247	0.184	0.138	0.104	0.085	0.073
3.616E-5	0.0	1.062E-4	0.0	6.566	1.0		
0.889	1.045	1.166	1.195	1.172	1.112	1.015	0.888
0.759	0.668	0.630	0.631	0.640	0.641	0.620	0.567
0.489	0.389	0.297	0.221	0.166	0.126	0.103	0.088

Serie ES.CO1N.3 (continuación)

3.616E-5	0.0	1.254E-4	0.0	6.558	1.0		
1.043	1.226	1.365	1.399	1.376	1.311	1.201	1.051
0.899	0.789	0.748	0.748	0.762	0.765	0.741	0.678
0.579	0.463	0.354	0.264	0.195	0.147	0.116	0.099
3.616E-5	0.0	1.433E-4	0.0	6.524	1.0		
1.182	1.392	1.556	1.600	1.573	1.495	1.363	1.185
1.006	0.877	0.829	0.827	0.845	0.848	0.823	0.755
0.646	0.519	0.397	0.295	0.219	0.166	0.131	0.111
3.616E-5	0.0	1.612E-4	0.0	6.560	1.0		
1.292	1.520	1.707	1.762	1.736	1.644	1.491	1.285
1.079	0.931	0.875	0.873	0.892	0.899	0.872	0.801
0.687	0.551	0.422	0.314	0.233	0.176	0.138	0.116
3.616E-5	0.0	1.792E-4	0.0	6.507	1.0		
1.392	1.633	1.846	1.918	1.898	1.801	1.616	1.371
1.130	0.956	0.892	0.888	0.906	0.914	0.887	0.816
0.702	0.565	0.435	0.326	0.243	0.186	0.147	0.125

Serie ES.CO1N.4

7.232E-5	0.0	1.054E-4	0.0	3.985	1.0			
0.546	0.622	0.768	0.866	0.904	0.851	0.693	0.473	
0.272	0.120	0.060	0.040	0.032	0.027	0.022	0.019	
0.016	0.013	0.010	0.008	0.006	0.005	0.004	0.003	
7.232E-5	0.0	1.054E-4	0.0	4.271	1.0			
0.573	0.651	0.796	0.888	0.921	0.861	0.705	0.492	
0.299	0.153	0.096	0.074	0.064	0.055	0.048	0.042	
0.036	0.030	0.025	0.020	0.016	0.014	0.011	0.010	
7.232E-5	0.0	1.054E-4	0.0	4.560	1.0			
0.608	0.694	0.827	0.904	0.922	0.866	0.725	0.538	
0.367	0.239	0.185	0.157	0.142	0.126	0.113	0.098	
0.082	0.065	0.050	0.038	0.028	0.021	0.015	0.012	
7.232E-5	0.0	1.054E-4	0.0	4.823	1.0			
0.647	0.740	0.867	0.934	0.940	0.883	0.756	0.587	
0.432	0.316	0.260	0.230	0.214	0.200	0.185	0.165	
0.141	0.113	0.088	0.067	0.051	0.040	0.032	0.027	
7.232E-5	0.0	1.054E-4	0.0	5.085	1.0			
0.698	0.804	0.931	0.990	0.990	0.929	0.806	0.645	
0.495	0.384	0.328	0.300	0.285	0.272	0.255	0.229	
0.195	0.157	0.122	0.093	0.070	0.055	0.044	0.037	
7.232E-5	0.0	1.054E-4	0.0	5.243	1.0			
0.748	0.865	0.985	1.033	1.026	0.966	0.855	0.709	
0.570	0.467	0.417	0.393	0.382	0.371	0.353	0.321	
0.274	0.220	0.171	0.130	0.099	0.076	0.060	0.052	
7.232E-5	0.0	1.054E-4	0.0	5.562	1.0			
0.825	0.963	1.084	1.125	1.108	1.046	0.940	0.800	
0.665	0.564	0.519	0.503	0.502	0.495	0.476	0.435	
0.373	0.299	0.230	0.173	0.131	0.100	0.080	0.069	
7.232E-5	0.0	1.054E-4	0.0	5.843	1.0			
0.876	1.028	1.150	1.183	1.160	1.097	0.996	0.863	
0.731	0.635	0.595	0.587	0.593	0.593	0.572	0.523	
0.448	0.359	0.274	0.204	0.153	0.116	0.092	0.078	
7.232E-5	0.0	1.054E-4	0.0	6.045	1.0			
0.904	1.062	1.184	1.214	1.190	1.126	1.027	0.897	
0.763	0.669	0.630	0.627	0.635	0.637	0.616	0.563	
0.482	0.386	0.294	0.219	0.163	0.124	0.097	0.082	
7.232E-5	0.0	1.054E-4	0.0	6.202	1.0			
0.915	1.079	1.198	1.227	1.201	1.140	1.040	0.911	
0.778	0.684	0.646	0.646	0.657	0.660	0.640	0.585	
0.501	0.401	0.305	0.226	0.169	0.128	0.100	0.085	

Serie ES.CO1N.4 (continuación)

7.232E-5	0.0	1.054E-4	0.0	6.552	1.0		
0.937	1.103	1.226	1.252	1.225	1.163	1.067	0.939
0.806	0.712	0.677	0.680	0.693	0.698	0.677	0.618
0.528	0.422	0.321	0.237	0.177	0.133	0.105	0.089
7.232E-5	0.0	1.054E-4	0.0	6.860	1.0		
0.941	1.104	1.223	1.248	1.223	1.163	1.071	0.948
0.819	0.728	0.693	0.695	0.707	0.709	0.685	0.624
0.533	0.425	0.324	0.239	0.178	0.135	0.106	0.090
7.232E-5	0.0	1.054E-4	0.0	7.228	1.0		
0.944	1.106	1.225	1.250	1.226	1.167	1.077	0.955
0.828	0.738	0.703	0.705	0.716	0.716	0.691	0.629
0.536	0.427	0.325	0.240	0.178	0.135	0.106	0.090

Anexo 5.3: datos espectroscópicos del sistema Pd(II)-31N2MPSerie ES.PD1N.1

7.846E-6	0.0	1.964E-6	0.0	1.113	1.0			
0.028	0.024	0.025	0.023	0.024	0.024	0.024	0.024	0.023
0.023	0.022	0.020	0.017	0.016	0.014	0.014	0.014	0.013
0.012	0.010	0.009	0.009	0.008	0.007	0.006	0.006	0.005
7.846E-6	0.0	3.928E-6	0.0	1.113	1.0			
0.055	0.049	0.048	0.047	0.049	0.049	0.049	0.048	0.047
0.047	0.043	0.040	0.037	0.035	0.033	0.032	0.032	0.031
0.028	0.026	0.023	0.022	0.022	0.020	0.018	0.018	0.016
7.846E-6	0.0	7.856E-6	0.0	1.113	1.0			
0.095	0.084	0.082	0.082	0.085	0.087	0.087	0.087	0.085
0.082	0.076	0.070	0.063	0.059	0.056	0.054	0.054	0.051
0.048	0.044	0.040	0.036	0.034	0.031	0.027	0.027	0.024
7.846E-6	0.0	1.178E-5	0.0	1.113	1.0			
0.127	0.113	0.109	0.109	0.113	0.116	0.117	0.117	0.116
0.113	0.107	0.101	0.096	0.092	0.089	0.087	0.087	0.084
0.080	0.074	0.067	0.060	0.054	0.047	0.040	0.040	0.035
7.846E-6	0.0	1.571E-5	0.0	1.113	1.0			
0.153	0.137	0.136	0.140	0.148	0.152	0.152	0.152	0.145
0.136	0.124	0.113	0.106	0.101	0.096	0.094	0.094	0.091
0.086	0.080	0.073	0.067	0.061	0.053	0.046	0.046	0.040
7.846E-6	0.0	1.964E-5	0.0	1.113	1.0			
0.172	0.156	0.159	0.168	0.180	0.186	0.183	0.183	0.172
0.154	0.134	0.117	0.108	0.102	0.098	0.096	0.096	0.093
0.088	0.082	0.075	0.069	0.062	0.055	0.047	0.047	0.041
7.846E-6	0.0	2.357E-5	0.0	1.113	1.0			
0.193	0.177	0.183	0.196	0.212	0.221	0.216	0.216	0.198
0.173	0.145	0.124	0.111	0.105	0.101	0.099	0.099	0.095
0.091	0.084	0.077	0.070	0.063	0.056	0.048	0.048	0.042
7.846E-6	0.0	2.750E-5	0.0	1.113	1.0			
0.210	0.197	0.207	0.225	0.246	0.256	0.249	0.249	0.226
0.191	0.155	0.128	0.113	0.106	0.102	0.100	0.100	0.097
0.092	0.087	0.079	0.071	0.064	0.057	0.049	0.049	0.043
7.846E-6	0.0	3.142E-5	0.0	1.113	1.0			
0.239	0.221	0.234	0.255	0.280	0.291	0.282	0.282	0.253
0.212	0.170	0.138	0.120	0.112	0.106	0.103	0.103	0.099
0.094	0.088	0.080	0.073	0.066	0.058	0.050	0.050	0.044
7.846E-6	0.0	3.928E-5	0.0	1.113	1.0			
0.269	0.255	0.280	0.313	0.347	0.363	0.350	0.350	0.306
0.246	0.185	0.140	0.118	0.108	0.104	0.102	0.102	0.099
0.094	0.088	0.081	0.073	0.066	0.058	0.050	0.050	0.043

Serie ES.PD1N.2

9.807E-7	0.0	9.034E-6	0.0	1.113	1.0		
0.059	0.055	0.065	0.073	0.083	0.086	0.080	0.067
0.053	0.036	0.025	0.019	0.017	0.016	0.016	0.015
0.014	0.012	0.010	0.009	0.008	0.007	0.005	0.004
1.961E-6	0.0	8.249E-6	0.0	1.113	1.0		
0.062	0.056	0.062	0.068	0.075	0.077	0.074	0.065
0.053	0.041	0.031	0.027	0.024	0.023	0.023	0.022
0.021	0.019	0.016	0.014	0.013	0.010	0.008	0.006
2.942E-6	0.0	7.070E-6	0.0	1.113	1.0		
0.065	0.057	0.059	0.062	0.066	0.068	0.066	0.061
0.056	0.048	0.043	0.039	0.037	0.036	0.036	0.035
0.032	0.029	0.026	0.023	0.020	0.017	0.014	0.013
3.596E-6	0.0	6.481E-6	0.0	1.113	1.0		
0.070	0.061	0.061	0.063	0.066	0.067	0.065	0.063
0.060	0.055	0.051	0.047	0.045	0.043	0.043	0.042
0.039	0.036	0.032	0.028	0.025	0.022	0.018	0.015
4.250E-6	0.0	5.892E-6	0.0	1.113	1.0		
0.069	0.061	0.060	0.059	0.061	0.062	0.061	0.061
0.060	0.056	0.054	0.050	0.048	0.047	0.046	0.044
0.041	0.037	0.033	0.029	0.027	0.024	0.019	0.016
4.904E-6	0.0	5.303E-6	0.0	1.113	1.0		
0.067	0.059	0.058	0.057	0.060	0.061	0.060	0.059
0.058	0.054	0.049	0.045	0.042	0.039	0.038	0.036
0.034	0.030	0.028	0.025	0.024	0.022	0.018	0.016
5.557E-6	0.0	4.714E-6	0.0	1.113	1.0		
0.064	0.057	0.056	0.055	0.057	0.058	0.057	0.056
0.055	0.050	0.047	0.042	0.039	0.037	0.035	0.034
0.031	0.028	0.025	0.023	0.022	0.020	0.017	0.015
6.211E-6	0.0	4.124E-6	0.0	1.113	1.0		
0.053	0.047	0.048	0.045	0.047	0.048	0.047	0.046
0.045	0.041	0.038	0.034	0.032	0.032	0.030	0.028
0.026	0.022	0.020	0.019	0.018	0.016	0.013	0.012
7.192E-6	0.0	3.339E-6	0.0	1.113	1.0		
0.048	0.042	0.040	0.039	0.040	0.040	0.040	0.039
0.038	0.035	0.033	0.031	0.027	0.025	0.025	0.023
0.021	0.019	0.017	0.015	0.015	0.013	0.011	0.010

Serie ES.PD1N.2 (continuación)

8.173E-6	0.0	2.357E-6	0.0	1.113	1.0		
0.035	0.031	0.031	0.030	0.031	0.031	0.031	0.030
0.029	0.027	0.025	0.022	0.020	0.019	0.018	0.017
0.016	0.014	0.012	0.011	0.010	0.010	0.008	0.006
8.826E-6	0.0	1.375E-6	0.0	1.113	1.0		
0.023	0.020	0.019	0.018	0.018	0.018	0.017	0.017
0.017	0.016	0.015	0.013	0.012	0.011	0.011	0.010
0.009	0.007	0.006	0.005	0.005	0.004	0.003	0.002

Serie ES.PD1N.3

9.807E-6	0.0	2.696E-5	0.0	1.111	1.0		
0.225	0.210	0.222	0.241	0.263	0.273	0.264	0.236
0.199	0.160	0.130	0.114	0.106	0.101	0.099	0.095
0.090	0.084	0.077	0.071	0.065	0.057	0.049	0.044
9.807E-6	0.0	2.696E-5	0.0	1.392	1.0		
0.226	0.210	0.220	0.238	0.257	0.266	0.258	0.233
0.197	0.161	0.133	0.118	0.111	0.107	0.105	0.101
0.096	0.089	0.082	0.074	0.067	0.059	0.050	0.044
9.807E-6	0.0	2.696E-5	0.0	1.933	1.0		
0.226	0.210	0.220	0.237	0.256	0.263	0.255	0.229
0.195	0.159	0.133	0.119	0.112	0.108	0.105	0.101
0.096	0.090	0.082	0.074	0.066	0.058	0.049	0.043
9.807E-6	0.0	2.696E-5	0.0	2.183	1.0		
0.228	0.214	0.224	0.241	0.258	0.265	0.255	0.228
0.194	0.159	0.133	0.119	0.113	0.109	0.106	0.103
0.097	0.091	0.083	0.075	0.068	0.060	0.051	0.044
9.807E-6	0.0	2.696E-5	0.0	2.428	1.0		
0.229	0.216	0.227	0.245	0.262	0.268	0.256	0.227
0.193	0.158	0.132	0.117	0.111	0.106	0.103	0.099
0.093	0.086	0.079	0.071	0.064	0.057	0.049	0.042
9.807E-6	0.0	2.696E-5	0.0	2.866	1.0		
0.229	0.218	0.229	0.245	0.259	0.261	0.246	0.216
0.184	0.151	0.130	0.118	0.112	0.109	0.107	0.102
0.095	0.088	0.079	0.070	0.063	0.056	0.048	0.042
9.807E-6	0.0	2.696E-5	0.0	3.461	1.0		
0.230	0.220	0.235	0.250	0.260	0.256	0.234	0.203
0.172	0.145	0.130	0.122	0.115	0.108	0.101	0.092
0.084	0.075	0.066	0.059	0.053	0.047	0.040	0.035
9.807E-6	0.0	2.696E-5	0.0	3.872	1.0		
0.231	0.223	0.240	0.253	0.261	0.253	0.229	0.196
0.163	0.137	0.126	0.120	0.113	0.108	0.099	0.089
0.079	0.070	0.061	0.055	0.049	0.043	0.036	0.032
9.807E-6	0.0	2.696E-5	0.0	4.193	1.0		
0.229	0.223	0.240	0.253	0.260	0.252	0.227	0.192
0.162	0.138	0.127	0.122	0.116	0.110	0.101	0.092
0.081	0.071	0.063	0.055	0.050	0.044	0.037	0.033
9.807E-6	0.0	2.696E-5	0.0	5.571	1.0		
0.242	0.231	0.246	0.257	0.259	0.248	0.224	0.191
0.166	0.147	0.142	0.138	0.130	0.117	0.102	0.085
0.069	0.056	0.046	0.040	0.036	0.032	0.026	0.022

Serie ES.PD1N.4

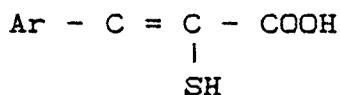
1.371E-5	0.0	2.935E-5	0.0	3.526	1.0		
0.264	0.252	0.265	0.280	0.291	0.288	0.268	0.236
0.202	0.171	0.154	0.143	0.137	0.130	0.124	0.117
0.107	0.098	0.088	0.079	0.072	0.063	0.055	0.048
1.371E-5	0.0	2.935E-5	0.0	3.663	1.0		
0.261	0.253	0.266	0.279	0.289	0.285	0.265	0.232
0.199	0.170	0.154	0.146	0.141	0.134	0.126	0.118
0.108	0.099	0.089	0.081	0.073	0.065	0.057	0.049
1.371E-5	0.0	2.935E-5	0.0	3.849	1.0		
0.262	0.252	0.265	0.278	0.287	0.281	0.259	0.225
0.194	0.167	0.152	0.144	0.137	0.131	0.125	0.117
0.108	0.098	0.089	0.081	0.073	0.065	0.057	0.050
1.371E-5	0.0	2.935E-5	0.0	4.233	1.0		
0.267	0.256	0.271	0.282	0.289	0.281	0.256	0.222
0.192	0.166	0.154	0.148	0.141	0.135	0.126	0.115
0.104	0.093	0.083	0.075	0.067	0.059	0.051	0.044
1.371E-5	0.0	2.935E-5	0.0	4.638	1.0		
0.269	0.259	0.274	0.287	0.292	0.284	0.257	0.223
0.192	0.168	0.155	0.149	0.142	0.132	0.123	0.110
0.097	0.086	0.075	0.067	0.060	0.053	0.046	0.041
1.371E-5	0.0	2.935E-5	0.0	5.964	1.0		
0.274	0.264	0.279	0.288	0.292	0.281	0.256	0.222
0.194	0.172	0.163	0.157	0.150	0.139	0.126	0.110
0.096	0.083	0.071	0.063	0.057	0.050	0.044	0.039

CONCLUSIONES

A) Sobre los reactivos estudiados:

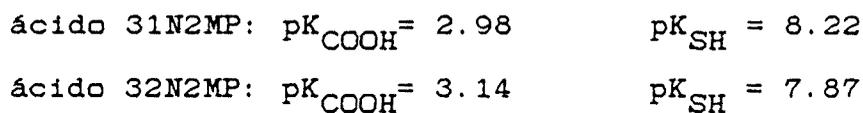
1. Se han preparado los ácidos 3-(1-Naftil)-2-mercaptopropenoico y 3-(2-Naftil)-2-mercaptopropenoico, según el procedimiento descrito por Campaigne y Cline, a partir de la rodanina y los aldehídos correspondientes.
2. Se han establecido métodos adecuados para evaluar la pureza de los mercaptoácidos. En los métodos empleados se ha aprovechado el carácter reductor del grupo mercapto (mediante valoración iodométrica) y el carácter ácido de los grupos carboxílico y mercapto (mediante valoración potenciométrica con NaOH).
3. Previa purificación de los reactivos, se ha realizado su análisis elemental y se han determinado algunas de sus características: solubilidad en diferentes disolventes, espectros de IR, RMN y UV-visible, estabilidad en disolución, y se han determinado sus constantes de ionización por métodos espectrofotométricos.
4. En el espectro infrarrojo de estas sustancias se ha observado, entre otras, la banda de absorción característica del enlace S-H, mostrando que en forma sólida se encuentran en forma tiol.

5. Los espectros RMN muestran diferencias entre los dos reactivos, destacando el diferente desplazamiento del protón del grupo -SH, en acetona deuterada.
6. Los espectros UV muestran la banda característica del sistema conjugado:



En disolución acuosa, se observa un desplazamiento hipso-crómico de esta banda al ionizarse el grupo carboxilo, mientras que la ionización del grupo mercapto da lugar a un desplazamiento batocrómico, de mayor amplitud que el anterior.

7. Los ácidos 3-Naftil-2-mercaptopropenoicos son compuestos poco estables en disolución, siendo más estable el derivado 1-Naftil que el 2-Naftil.
8. Se han determinado espectrofotométricamente las constantes de disociación de los mercaptoácidos en medio acuoso, a fuerza iónica 0.1 M, y temperatura de 25°C. Los resultados obtenidos han sido:



9. Se ha realizado un estudio cualitativo de la reaccionabilidad de estos mercaptoácidos frente a los iones metálicos en diferentes medios. De este estudio cabe destacar:

a) La analogía de las reacciones entre los dos Naftil-derivados y su similitud con las de otros ácidos 3-aril-2-mercaptopropenoicos estudiados previamente en este Departamento.

b) Las reacciones que han presentado mayor sensibilidad han tenido lugar con los iones: Cu(II), Pd(II), Mo(VI), Fe(II-III), Co(II), Ni(II), Mn(II) y Ti(IV). La reacción con Ti(IV) es característica de este tipo de compuestos, pues no la dan los mercaptoácidos saturados.

c) Gran parte de los complejos solubles e insolubles con los iones metálicos son extraíbles en disolventes orgánicos, aumentando en algunos casos la sensibilidad de las reacciones. La extracción se efectúa preferentemente con los disolventes oxigenados más polares, como el alcohol isoamílico y la metil-isobutil-cetona.

B) Estudios llevados a cabo sobre equilibrios en solución mediante técnicas potenciométricas

10. Como aportación al estudio de constantes de formación de complejos mediante valoraciones potenciométricas con

electrodos de pH, se ha diseñado y puesto en funcionamiento un entorno de programación alrededor de un ordenador personal HP 9816 S y sus periféricos.

El sistema desarrollado cuenta con tres programas, que tienen las funciones respectivas de:

- control automático de las valoraciones potenciométricas, y creación de los ficheros de datos una vez terminadas.
- cálculo y refine de las constantes de estabilidad de los sistemas que se estudian.
- representación gráfica de la distribución de las especies determinadas, en función del pH de la disolución.

11. Para comprobar el buen funcionamiento de los programas de automatización de valoraciones potenciométricas, y de cálculo de constantes de estabilidad, se han aplicado al estudio del sistema Ni(II)-glicina, en medio acuoso y fuerza iónica 1 M en NaCl, a la temperatura de 25°C. Los valores de las constantes de protonación del ligando y de complejación con ión Ni(II) que se han determinado son:

$$\begin{array}{lll} \log \beta_{101} = 9.66 & \log \beta_{102} = 12.07 & \\ \log \beta_{110} = 5.62 & \log \beta_{210} = 10.35 & \log \beta_{310} = 13.71 \end{array}$$

Los valores hallados han sido concordantes con los hallados por otros autores.

12. Se han determinado las constantes de protonación de los mercaptoácidos y de formación de complejos solubles con los iones Ni(II) y Zn(II), aplicando el sistema de valoración automática, y el programa de cálculo desarrollado en esta memoria para la determinación de las diferentes constantes de formación. El estudio se ha llevado a cabo en medio mixto agua-etanol al 50 % (v/v), a una temperatura de 25°C y fuerza iónica 1 M.
13. Del calibrado de los electrodos y el estudio del medio iónico utilizado, se ha comprobado la buena respuesta del electrodo de vidrio en medio ácido, y una cierta imprecisión en medio alcalino, aunque no se han presentado desviaciones sistemáticas. Se ha calculado un valor de pKw del medio de 14.25. En medio ácido, y con el sistema electrodo empleado, se ha encontrado un potencial de unión líquida de 13.7 mV/mol H⁺. En medio básico la unión líquida no ha sido apreciable.
14. Las constantes de protonación calculadas de los ácidos 3-Naftil-2-mercaptopropenoicos han sido:

$$\text{ácido 31N2MP: } \log \beta_{101} = 9.30 \qquad \log \beta_{102} = 13.37$$

$$\text{ácido 32N2MP: } \log \beta_{101} = 8.89 \qquad \log \beta_{102} = 13.05$$

15. Los complejos formados por estos mercaptoácidos con los iones Ni(II) y Zn(II) han dado lugar a especies mononucleares en el intervalo de concentraciones estudiado. En todos los casos se han encontrado especies de composición 1:1:0 y 2:1:0. En el sistema Zn(II)-32N2MP se ha determinado, adicionalmente, la especie mixta 1:1:1. Los valores de las constantes de formación son:

ácido 31N2MP:

$$\text{Ni}^{2+} : \log \beta_{110} = 9.16 \quad \log \beta_{210} = 19.96$$

$$\text{Zn}^{2+} : \log \beta_{110} = 9.68 \quad \log \beta_{210} = 18.89$$

ácido 32N2MP:

$$\text{Ni}^{2+} : \log \beta_{110} = 9.84 \quad \log \beta_{210} = 19.99$$

$$\text{Zn}^{2+} : \log \beta_{111} = 12.79 \quad \log \beta_{110} = 9.70 \quad \log \beta_{210} = 19.04$$

C) Equilibrios de complejación estudiados por métodos espectrofotométricos

16. Se han estudiado espectrofotométricamente los complejos formados por el ácido 31N2MP y los iones Ni(II), Pd(II) y Co(II), en medio acuoso a 25°C y fuerza iónica 0.1 M. La determinación de las constantes de equilibrio se ha llevado a cabo con el programa SQUAD, dando los resultados

siguientes:

$$\text{Ni}^{2+} : \log \beta_{110} = 8.10 \quad \log \beta_{210} = 16.35$$

$$\text{Pd}^{2+} : \log \beta_{110} = 15.56 \quad \log \beta_{210} = 26.40$$

$$\text{Co}^{2+} : \log \beta_{110} = 7.17 \quad \log \beta_{210} = 14.41 \quad \log \beta_{310} = 21.40$$

La formación de la especie 3:1:0 entre el ácido 31N2MP y el ión cobalto es debida, posiblemente, a una oxidación de este ión por el oxígeno del aire para dar complejos de Co(III).

APENDICE I:

DISEÑO Y PUESTA A PUNTO DE UN SISTEMA AUTOMATICO DE VALORACIONES POTENCIOMETRICAS

I.1. Introducción

Uno de los métodos más ampliamente usados en el estudio de equilibrios de formación de complejos en solución entre un ión metálico y un ligando es el basado en la valoración potenciométrica de una disolución ácida, que contiene ligando y metal, con una base fuerte, y en la que se sigue la concentración de ión hidrógeno libre en función del volumen de valorante añadido (método de Calvin y Wilson, ref. 1) mediante dos electrodos: uno de referencia (generalmente de Ag/AgCl o bien de calomelanos saturado) y un electrodo sensible a los iones hidrógeno, que puede ser el electrodo de hidrógeno (H_2/H^+), el de quinhidrona (mezcla equimolecular de quinona e hidroquinona) o bien el electrodo de vidrio. Por razones prácticas en la experimentación este último es el mayoritariamente empleado cuando no se trabaja en las zonas más extremas de pH.

El método de valoración potenciométrica permite obtener

una gran cantidad de información, ya que a cada par de valores de volumen de base-pH de la disolución corresponde un punto de equilibrio en la valoración, por lo que con una serie de valoraciones en las que se varíen las concentraciones de metal y de ligando se puede obtener la cantidad de información suficiente para definir un sistema y poder cuantificar el equilibrio entre las especies reaccionantes.

La misma concepción del método potenciométrico de valoración permite tener un sistema automatizado de adquisición de datos, dado que la adición de un determinado volumen de valorante como la lectura del instrumento de medida (en pH o mV) es una tarea repetitiva, que no necesita que la medida sea extremadamente rápida, ya que los intervalos de lectura entre punto y punto generalmente son de varios minutos. Las características que presenta la valoración potenciométrica hacen aconsejable en lo posible su automatización, evitando de esta manera el estar pendiente del instrumento durante las 3-5 horas que como mínimo puede durar una valoración.

En la actualidad, el desarrollo de los microprocesadores y microcomputadores (y sobre todo de técnicas de programación) permite la fácil automatización del proceso potenciométrico.

Hasta hace pocos años, prácticamente el lenguaje utilizado para programar un sistema de control automático era el assembler (2, 3), lo cual hacía que un programa apareciera poco legible para un profano en la materia. Sin embargo, hoy día, en que la microinformática tiene una amplia base educativa, unido al lanzamiento de "hardware" a bajo precio hace que no sea difícil su adaptación a problemas particulares como el que se nos plantea. Por otro lado, los lenguajes de programación de alto nivel (BASIC, PASCAL, FORTRAN, etc.) implantados en los microordenadores, permiten el control de periféricos de manera mucho más legible que el assembler.

En el caso del estudio potenciométrico, el lenguaje más apropiado en principio es el BASIC, especialmente cuando está residente en la memoria como intérprete (no compilado) que permite un enfoque interactivo. Un lenguaje BASIC interpretado es más lento cuando se le compara con un lenguaje compilado, pero como hemos señalado, en una valoración potenciométrica no es necesario tener una gran rapidez de respuesta (Para una comparación de los diferentes lenguajes de programación, ver ref. 4).

Sin embargo, en nuestro trabajo no se ha utilizado ningún sistema del tipo indicado, ya que se disponía de un microordenador de los llamados "supermicroordenadores" (HP 9816).

En la bibliografía reciente se encuentran descritos varios sistemas de valoración automática basados en microprocesador o microcomputador (5, 6, 7), de los que comentaremos brevemente los dos últimos:

Zuberbuhler y col. (6) utilizan un sistema basado en microprocesador, el cual almacena los datos en una cinta magnética, y mediante el uso de un programa auxiliar son pasados a otra cinta para poder ser utilizado en un microordenador HP 9835. El tratamiento numérico de estos datos puede ser llevado a cabo en el mismo ordenador mediante el programa TITFIT (8). Arnold y col. (7) controlan la valoración a través de un microordenador IBM-PC, siendo uno de los sistemas más sofisticados, ya que permite valoraciones con incrementos de volumen constante, o bien pueden ser variados automáticamente según el programa para obtener unos incrementos de potencial constantes entre dos puntos consecutivos de la valoración. De esta manera los autores llegan a determinar la concentración de ión carbonato presente como impureza en la disolución de base valorante. El almacenamiento de los datos en este caso es totalmente "on line", ya que posteriormente son llevados al disco de trabajo del ordenador. El tratamiento de los datos se puede efectuar mediante una versión de MINIQVAD adaptada al IBM-PC. La única limitación es que por cuestiones de memoria y sistema operativo sólo pueden ser tratadas las valoraciones una por una.

En el presente trabajo se han desarrollado de manera conjunta y complementaria dos programas: uno para la valoración automática y almacenamiento de los datos y un programa de tratamiento riguroso de estos datos, el cual se presenta en el apéndice II (programa MINIGLASS).

El diseño de un sistema completo así planteado permite facilitar al experimentador dos de las partes más tediosas en el estudio potenciométrico de equilibrios en disolución, la de la valoración propiamente dicha y la de la transcripción de los datos al ordenador para su tratamiento. Además, los errores (siempre presentes) al escribir los datos -tales como el poner un punto o coma de más o de menos, olvidarse de escribir un número clave, etc.- son en buena parte evitados de esta manera.

Adicionalmente también se ha desarrollado un programa auxiliar para la representación gráfica de los diagramas de distribución de especies de los complejos obtenidos frente al pH de la disolución (programa DISTR, en apéndice III).

El buen funcionamiento del sistema de valoración y del programa de tratamiento de los datos se ha comprobado aplicándolos al estudio del sistema Ni(II)-glicinato. La elección de este sistema obedece en primer lugar, a que fue objeto por parte del grupo italiano de "Termodinamica dei Complessi" de un estudio interlaboratorio de técnicas de

valoración y tratamiento de resultados (9, 10) y por otra parte porque ha sido análogamente utilizado para comprobar la aplicación de otros programas, como TITFIT (8) y SUPERQUAD (11), así como en el sistema de valoración automática previamente comentado (7), lo que ha permitido tener a nuestro alcance una amplia bibliografía reciente para poder comparar el sistema desarrollado.

1.2. Programa de Adquisición de Datos (AUTO)

Este programa está escrito en BASIC (H.P. 2.0 con extensiones 2.1), y permite el control automático de la valoración. El listado que se adjunta corresponde a la versión que controla una sola valoración, pero dado su carácter modular, la ampliación necesaria para poder controlar varias valoraciones simultáneamente es muy sencilla dadas las facilidades del sistema operativo que emplea el ordenador utilizado.

El soporte de "hardware" empleado consiste de las siguientes unidades:

- unidad central HP 9816 S , con 768 Kbytes de memoria RAM .
- almacenamiento de memoria externo HP 9134XV , consistente en una unidad de disco flexible de 3 1/2 pulgadas (270 Kbytes) y un disco fijo tipo Winchester de 15 Mbytes de memoria.
- impresora HP 82906 A.
- unidad de adquisición de datos HP 3421A , equipada con un multiplexor de ocho canales de entrada analógica (opción 020) y una salida digital de ocho canales (opción 050).

Los periféricos están conectados a la unidad central a través de la interface en paralelo "HP-IB" (Hewlett - Packard Interface Bus) , equivalente al estándar IEEE-488.

I.2.1. Funcionamiento del programa

La unidad de adquisición de datos HP 3421 A es la encargada de leer el pH-metro y del control de las buretas automáticas. Las lecturas de potencial se realizan a través de la salida de registrador del pH-metro, conectada con la entrada analógica 020, señal que a su vez es convertida a señal digital y enviada al ordenador por la interace HP-IB para su lectura. Las salidas digitales de la unidad (opción 050) activan el motor de la autobureta según las órdenes recibidas de la unidad central; de hecho, estas salidas digitales consisten en 8 interruptores contruidos según la tecnología MOS-FET (transistores de efecto de campo) que corresponden a un interruptor abierto o cerrado según se encuentren en estado de alta o baja impedancia.

Se podría objetar que la medida de pH o mV propiamente dicha es algo complicada, ya que primero es leída una señal analógica, posteriormente transformada a digital y finalmente enviada a la unidad central, puesto que la mayoría de los pH-metros actuales tienen ya una salida digital para ordenador que puede ser aprovechada. Sin embargo, comparando las dos maneras de tomar los datos, de hecho existen los mismos pasos, ya que los pH-metros digitales consisten en un amplificador de señal que divide la señal amplificada en dos partes, una que va directamente a la salida de registrador, y otra que a su vez pasa por un convertidor A/D y que finalmente llega al

"display" del pH-metro y es enviada además por la salida digital, por lo que el número de transformaciones que sufre la señal enviada por el electrodo es la misma.

Por otra parte, las salidas digitales del pH-metro tienen el inconveniente de que no todos envían la misma, ya que puede ser del tipo RS 232 C (por ejemplo, los CRISON), BCD (ORION y RADIOMETER), etc.

La solución empleada por nosotros presenta la ventaja de que al ser analógica y directamente proporcional a la medida del pH-metro, el único cambio que puede llegar a ser necesario es una pequeña corrección (por ejemplo, en los RADIOMETER dividimos la señal obtenida por un factor de 5, y en ORION se divide por 2.507) muy fácil de determinar. Según nuestra experiencia, en los modelos de pH-metros ORION 701, RADIOMETER PHM 64 y PHM 84, las desviaciones máximas observadas entre la lectura del "display" del medidor y la recibida por el ordenador han sido siempre menores de 0.2 mV, que corresponde al propio error de este tipo de pH-metros.

Hay que resaltar también el hecho de que con el sistema que se presenta es posible controlar hasta 8 valoraciones simultáneamente sin modificación de "hardware" y con muy pocos cambios en el programa descrito a continuación.

1.2.2. Descripción del programa

Las especificaciones de las líneas de programa se describen a continuación brevemente comentadas:

- líneas 10-20: en ellas se dimensionan las matrices numéricas y alfanuméricas en las que son almacenados los datos.

- líneas 60-90: se pregunta al usuario qué nombre le va a dar al archivo de datos y si va a ir dirigido al disco duro (K=1) o al disco flexible (K=2). Si el usuario se ha equivocado al asignar el valor de K (no es ni 1 ni 2) se lo vuelve a preguntar. La línea 90 pone a 0 el contador de puntos experimentales .

- líneas 100-180: pide al usuario información sobre los parámetros esenciales de la curva de valoración: volumen inicial, E_0 y g del electrodo (la E_0 no hace falta que sea totalmente exacta, con un valor aproximado es suficiente, ya que seguramente será refinado por MINIGLASS) y concentración de valorante. A continuación pide el número de componentes del sistema y la concentración inicial de cada uno. A la vez que el programa va recibiendo esta información almacena estos valores como elementos de la matriz de datos "Exper".

- línea 190: define una tecla de función que aparecerá siempre en la pantalla del ordenador con el rótulo "FIN" . Debido a que este tipo de instrucciones siempre tiene prioridad sobre el resto de instrucciones ejecutadas por el programa, un toque en esta tecla de función nos envía directamente a la subrutina "Final" comentada posteriormente.

- líneas 210-250: la primera envía a la impresora una línea de escritura, mientras que las siguientes ponen a cero una serie de variables que son utilizadas durante el programa.

- líneas 260 y 270: envían a la unidad de adquisición de datos las órdenes de lectura de potencial (DCV: indica que se van a medir Voltios de Corriente Continua) y recogida de la señal asignándola a la variable "E". En este caso en particular se lee el canal No. 4 (del multiplexor). El valor de "709" corresponde a códigos propios de H.P., que nos dice que empleamos una interface HP-IB (tiene el código 7..) y que a la unidad de adquisición de datos le corresponde el número 9 (la interface HP-IB soporta simultáneamente hasta 14 periféricos), por tanto se obtiene el número 709.

- líneas 280-310: en este caso la lectura se toma de un pHmetro ORION 701 , y por tanto a la lectura se le efectúa una corrección previa antes de ser tratada. Después es

impresa en pantalla.

- líneas 320-420: comparan la lectura obtenida con la lectura anterior. Si la diferencia entre ellas es menor que 0.1 mv, el par de valores V_t/E y el tiempo que ha tardado en estabilizarse el sistema (en segundos) son transmitidos a la impresora, y los datos almacenados en la matriz "Exper". El contador de puntos experimentales se incrementa en una unidad. Si la diferencia es mayor que 0.1 mV se espera hasta hacer una nueva lectura (normalmente se espera un minuto y medio entre cada una). Por otra parte, si se han tomado un número de lecturas mayor que un valor determinado (por ejemplo 6, en este caso) se toma la última lectura y se continúa la ejecución del programa. Todo esto se ve reflejado en el listado que da la impresora, ya que nos dice el tiempo que ha tardado en estabilizarse, y de ahí podemos saber cuáles han sido los puntos más inestables.

- líneas 430-470: estas líneas se han previsto como opción para controlar la cantidad de valorante añadido cada vez. Las buretas Metrohm utilizadas permiten (en el caso que se utilice el cilindro de 10 ml o de 20 ml) una adición de 0.1 ml de valorante por cada "pulso" enviado a la bureta. En estas líneas se pueden programar el número de pulsos (la variable 'Incr') que son enviados a la bureta por medio de la salida digital. En el caso particular que se

representa en el listado quiere decir que si el volumen de valorante total añadido (V_t) es menor o igual que 15 ml, las adiciones serán de 0.5 ml (0.1×5), y que una vez pasado este volumen serán de 0.1 ml.

- líneas 480-540: controlan la bureta de acuerdo con los pulsos que se deben dar en cada adición. Consiste en efectuar un ciclo que envía cada vez un pulso a la bureta, y a la vez incrementa en 0.1 ml la variable ' V_t '; esto es repetido ' $Incr$ ' veces. El control de la bureta propiamente dicho es a través de las instrucciones ' CLS ' y ' OPN ' que consisten en "cerrar" o "abrir" el circuito de conexión. En este caso, la bureta está conectada a la salida No. 12 de la unidad HP 3421 A.

- línea 550: una vez efectuada la adición de valorante, la variable ' $Cont$ ', que indica el número de lecturas tomadas de un punto hasta que se estabiliza, es puesta a cero.

- línea 580-610 (subrutina Espera): es donde se toma el intervalo entre cada lectura de potencial. En el caso presentado se toma un tiempo de 90 segundos antes de continuar la ejecución del programa, para ello realiza un bucle 90 veces que espera un segundo cada vez (por la instrucción ' $WAIT 1$ ').

- líneas 620-700 (subrutina Final): son las líneas a donde se envía la ejecución del programa cuando se apreta la tecla de función definida "FIN". Primero asigna al primer elemento de la matriz 'Exper' el número total de puntos experimentales que hay, y después redimensiona la matriz al valor mínimo posible. Finalmente envía un archivo de datos al dispositivo de memoria externa elegido. Una vez hecho esto, se le pregunta al usuario si quiere continuar la valoración, y en caso afirmativo se continúa ejecutando el programa donde se encontraba antes de tocar la tecla FIN. En caso contrario para el programa. Para evitar que el usuario toque accidentalmente la tecla FIN, al iniciar esta subrutina se le pregunta si en efecto está seguro de que quiere acabar. Si decide que aún no quiere enviar el archivo de datos, la ejecución del programa continúa normalmente.

Como se puede deducir de las entradas (inputs) en las líneas 100 a 180, la ordenación de los archivos de datos que efectúa este programa es la siguiente:

- número de puntos experimentales.
- volumen inicial de la disolución.
- E_0 y g de los electrodos.
- concentración de la solución valorante (positiva si es ácido y negativa si es base).
- concentración inicial de cada componente en la disolución, teniendo en cuenta que el ión hidrógeno siempre ocupa el último lugar.
- pares de valores V_{t_i}, E_i .

En el anexo correspondiente al apéndice II se presenta el conjunto de datos experimentales obtenidos en el estudio Ni(II)-glicina, donde se puede ver de manera práctica la estructura de los archivos de datos.

1.2.3. Listado del programa AUTO

```

10  OPTION BASE 1
20  DIM Exper(200)
30  DIM Nombre$(10),Msus$(2)(12),Bout$(12)
40  Msus$(1)=":HP913X,700"
50  Msus$(2)=":HP8290X,702"
60  INPUT "NOMBRE DEL ARCHIVO Y DISCO (1:Winchester, 2:Floppy)",
      Nombre$,K
70  IF K<>1 AND K<>2 THEN 60
80  Bout$=Msus$(K)
90  Npuntos=0
100 INPUT "          Volumen inicial (ml) ?",Exper(2)
110 INPUT "          Eo y G del electrodo?",Exper(3),Exper(4)
120 INPUT "  Concentración de valorante (+:ácido , -:base)?",
      Exper(5)
130 INPUT "          Número de componentes?",Ncom
140 FOR Z=1 TO Ncom
150 PRINT TABXY(10,22);"  Concentración del componente ";Z
160 INPUT Exper(5+Z)
170 NEXT Z
180 L=5+Ncom
190 ON KEY 5 LABEL "  FIN " GOSUB Final
210 OUTPUT 701;" Vt  E1  tiempo de estabilizar(SEG.)",
220 Cont=0
230 E=0
240 Vt=0
250 E2=0
260 OUTPUT 709;"DCV";4
270 ENTER 709;E
280 E=-1000*E/2.506
290 E=.01*INT(100*E+.5)
310 PRINT E
320 IF ABS(E-E2)<.1 THEN GOTO 370
330 E2=E
340 Cont=Cont+1
350 IF Cont>6 THEN GOTO 370
360 GOTO 560
370 !
380 OUTPUT 701 USING "DD.DD,XXXX,DDDDD.DD,XXX,DDD";Vt;E;
      (Cont+1)*90
390 Exper(L+1)=Vt
400 Exper(L+2)=E
410 L=L+2
420 Npuntos=Npuntos+1

```

```
430 IF Vt<=15 THEN
440     Incr=5
450 ELSE
460     Incr=1
470 END IF
480 FOR I=1 TO Incr
490     OUTPUT 709;"CLS12"
500     WAIT .5
510     Vt=Vt+.1
520     OUTPUT 709;"OPN12"
530     WAIT 1
540 NEXT I
550 Cont=0
560 GOSUB Espera
570 GOTO 260
580 Espera:FOR Dos=1 TO 90
590     WAIT 1
600     NEXT Dos
610     RETURN
620 Final: INPUT "     Estás seguro (S/N) ?",J$
630     IF J$="N" THEN 700
640         Exper(1)=Npuntos
650         REDIM Exper(L)
660         CREATE ASCII Nombre$&Bout$,5
670         ASSIGN @Path TO Nombre$&Bout$
680         OUTPUT @Path;Exper(*)
690         DISP "     ARCHIVO ";Nombre$&Bout$;" CONSTRUIDO"
700         INPUT " Quieres acabar (S/N)",J$
710         IF J$="S" THEN STOP
720         RETURN
730 END
```


I.3. BIBLIOGRAFIA

1. M. Calvin, K. W. Wilson; J. Amer. Chem. Soc., 67(1945)2003.
2. E. D. Salin, M. W. Blades, G. Horlick; Talanta, 28(1981)519.
3. M. Valiente; Tesis Doctoral, Barcelona 1981.
4. R.L.A. Sing, S.W. McGeorge, E.D. Salin; Talanta, 30(1983)805.
5. G. Velinov, N. Todorov, S. Karamphilov; Talanta, 30(1983)687.
6. A.D. Zuberbuhler, T.A. Kaden, H. Gamp, H. Mader; Talanta, 27(1980)513.
7. A. P. Arnold, S.A. Daignault, D.L. Rabenstein; Anal. Chem., 57(1985)1112.
8. A.D. Zuberbuhler, T.A. Kaden; Talanta, 29(1982)201.
9. E. Bottari, A. Braibanti, L. Ciavatta, A.M. Corrie, P. G. Daniele, F. Dallavalle, M. Grimaldi, A. Mastroianni, G. Mori, G. Ostacoli, F. Paoletti, E. Rizzarelli, S. Sammartano, C. Severini, A. Vacca, D. R. Williams; Ann. Chim. (Roma), 68(1978)813.
10. A. Braibanti, F. Dallavalle, G. Nori, B. Veroni; Talanta, 29(1982)725.
11. P. Gans, A. Sabatini, A. Vacca; J. Chem. Soc., Dalton Trans., (1985)1195.

APENDICE II:

DESARROLLO DE UN PROGRAMA DE DETERMINACION DE CONSTANTES DE EQUILIBRIO A PARTIR DE DATOS POTENCIOMETRICOS

II.1. Introducción

El programa MINIGLASS ha sido escrito y desarrollado durante la realización de la presente memoria para refinar los valores obtenidos en el estudio de las constantes de formación de equilibrios en disolución. Ha sido diseñado expresamente para su uso en el ordenador personal HP 9816 S, pero la adaptación a otros ordenadores no presenta problemas de consideración.

La mayoría de los programas más ampliamente usados en el cálculo de constantes de estabilidad según técnicas potenciométricas, tales como LETAGROP (1), SCOGS (2), MINIQVAD (3) o DALSF EK (4), han sido diseñados para su uso en grandes ordenadores ("mainframes"), debido a la gran cantidad de información necesaria en el cálculo de las constantes que mejor definen un sistema. Por otra parte, y debido a los avances en los últimos años en cuanto a capacidad de memoria y rapidez de cálculo de los microordenadores o computadores personales, han ido apareciendo paralelamente algunos programas de cálculo expresamente diseñados para éstos. Uno de los

pioneros en este tipo de programas fue el MINIPOT (5), escrito en BASIC para un microordenador de 16 Kbytes de memoria. Sin embargo, el ser un programa sencillo para ordenadores limitados, no permite el tratamiento de equilibrios de complejación seguidos por electrodos de pH, y además sólo permite el cálculo de una valoración cada vez. Por otra parte, uno de los principales méritos de este programa es su sencillez de utilización por su enfoque interactivo. Otros programas de interés para microordenadores son TITFIT (6) y BEST (7).

El programa BEST está escrito en FORTRAN para un microordenador de 64 Kbytes de memoria, y presenta como innovación un algoritmo para la resolución de los balances de masa totalmente general en cuanto al número de componentes y su caracterización (no hay la distinción particular entre "metal" y "ligando"). Por su parte, y debido a la memoria limitada, está dimensionado como máximo para unos 120 puntos experimentales en total.

TITFIT es un programa escrito en BASIC, y que tiene mayor capacidad que los anteriores, ya que permite tratar hasta 400 puntos experimentales y varias valoraciones simultáneamente, presentando como principal característica el emplear derivadas analíticas en su algoritmo de minimización. Esto es destacable, ya que de los programas citados hasta ahora, sólo MINIQVAD presenta esta característica. Esta particularidad le hace ser un programa de rápida ejecución, especialmente cuando se tiene en cuenta que emplea un lenguaje de programación

interpretado, si bien está limitado a tres componentes.

Cuando nos fijamos en el funcionamiento particular de los programas citados, vemos que la función a minimizar en cada uno puede ser muy diferente, por ejemplo en DALSF EK y MINIPOT es la f.e.m. medida, en SCOGS y TITFIT es el volumen de valorante añadido, en BEST es el pH, las concentraciones analíticas totales para cada componente de la disolución en MINIQUAD, y en LETAGROP se tiene a elección una serie de funciones como la f.e.m., la concentración analítica de ión hidrógeno o la \bar{n} .

Si bien un programa ejecutado en un gran ordenador será más rápido por su mayor capacidad que al ejecutarlo en un microordenador, éste puede ser utilizado para una interacción directa entre el usuario y el programa. En nuestra opinión esto es de gran importancia cuando para un sistema determinado se estudian diversos modelos, ya que permite comprobar "in situ" su ajuste a los datos experimentales. Los resultados obtenidos son más fácilmente asimilados por el usuario cuando éstos son presentados en forma de gráficos, por ejemplo, el programa TITFIT genera en la pantalla del ordenador las gráficas correspondientes a las curvas de valoración (la teórica y la experimental) y el diagrama de distribución de especies.

El programa desarrollado durante la presente tesis

presenta varias características no encontradas en los anteriores. En primer lugar, se ha escrito en PASCAL, un lenguaje de programación altamente estructurado y más avanzado que los comúnmente usados BASIC o FORTRAN. De hecho, una primera versión de este programa fue escrita en el BASIC ampliado de H.P., teniendo un funcionamiento altamente satisfactorio en general, pero presentaba el defecto de ser lento (por ser un lenguaje interpretado, a pesar de ser el HP BASIC uno de los más rápidos existentes), fundamentalmente debido al uso de derivadas numéricas en el proceso iterativo. Esta lentitud, y el hecho de disponer en nuestro ordenador de un sistema operativo y compilador de PASCAL nos indujeron a reescribir el programa en este lenguaje.

Otra de las características diferenciadoras del programa, es que reconoce tres tipos de valoraciones en las que los cálculos son efectuados mediante subrutinas diferentes: 'Gran' para el calibrado de electrodos y determinación del pK_w , 'Pkas' específico para el estudio de disociación ácido-base, y 'Complex' para el cálculo de constantes de formación de complejos. Ha sido diseñado de esta manera (es decir, una subrutina específica para cada tipo de problema) para que los cálculos en los balances de materia o carga sean lo más rápido posible.

Para que el programa sea de manejo sencillo, su funcionamiento es a base de "menús" que guían al usuario de una manera totalmente interactiva, ya que cuando se desea realizar una

tarea específica sólo hay que tocar una tecla determinada, y cuando hay que entrar datos numéricos los pregunta el mismo programa.

A diferencia de otros programas de cálculo en los que sólo se pueden refinar las constantes de estabilidad, MINIGLASS permite ajustar además los valores de E_0 y g de los electrodos, así como las concentraciones iniciales de cada componente en la cubeta de valoración, y la concentración de la disolución valorante (este último puede ser ácido ó alcali). De esta manera es posible refinar simultáneamente, por ejemplo, los valores de las constantes de formación y de las diferentes E_0 de cada valoración. Por su parte, el hecho de ser las concentraciones totales parámetros ajustables permite su uso en la determinación de errores sistemáticos, pureza de los reactivos, etc.

Las variables que se minimizan en este programa son, en los tres modos de funcionamiento (Gran, P_{Kas} y Complex), el logaritmo de la concentración de hidrógeno libre cambiado de signo (que para mayor sencillez, a lo largo de esta memoria vendrá indicado por $p(H)$). Adicionalmente, en el modo Complex la variable a minimizar puede ser ésta o bien el volumen de valorante añadido (V_t). La elección de la magnitud $p(H)$ como variable a ajustar en lugar del potencial medido por el electrodo (método más sencillo) obedece en primer lugar que al ser opcional en el procedimiento Complex las dos variables

$p(H)$ y V_t , en este último caso obliga a tener siempre en cuenta la concentración real (medida) de ión hidrógeno libre, y ya que debe ser conocida, es empleada para el refine según $p(H)$. En segundo lugar, y según nuestra experiencia, los residuales de V_t son sensiblemente del mismo orden que en $p(H)$, y por tanto comparables. Por otra parte, se obtienen prácticamente los mismos resultados refinando por $p(H)$ como por potenciales, ya que la relación entre sus desviaciones es siempre proporcional (recordemos que un error de 0.001 unidades de pH corresponde a un error en el potencial de 0.059 mV).

Respecto a este punto, hay que comentar que una de las características innovadoras del programa es precisamente, la definición del método empleado en el procedimiento Complex, puesto que permite el cambio de la variable a minimizar sin parar el programa, ya que es controlado mediante un indicador ("switch") accesible desde el mismo menú principal. Por comparación, recordemos que en LETAGROP se puede minimizar según distintas variables, que se especifican al principio de la ejecución del programa.

El enfoque ante todo interactivo del programa se pone de manifiesto en el apartado correspondiente a los gráficos. Para acceder a ellos hay que apretar la tecla correspondiente en el menú principal. Se generan dos tipos de gráficos: los de la curva (o curvas, si tratamos varias a vez) de valoración -la teórica frente a la calculada-, o bien de distribución de los

residuales (es decir, error frente a $p(H)$). El primer tipo nos sirve para ver si efectivamente el modelo teórico corresponde a la realidad en primera aproximación, mientras que la distribución de residuales es útil para ver qué tipo de error estamos cometiendo (por ejemplo, un error constante en todos los puntos nos indicará seguramente un potencial de electrodo incorrecto, etc.). Es de destacar que las rutinas empleadas para realizar el gráfico en la pantalla son las mismas que lo efectúan en el "plotter", y que la distinción entre una y el otro es mediante un indicador.

Los datos experimentales que trata el programa son leídos de dos tipos de archivos: archivos de datos propiamente dichos, y archivo 'modelo'. Los primeros se refieren a los datos correspondientes a cada valoración en particular (volumen inicial, potencial normal, concentraciones iniciales y de valorante, y los pares V_{t_i} , E_i), mientras que el archivo modelo contiene información común a todas las valoraciones (pK_w , potenciales de unión líquida, constantes de estabilidad, etc.). La organización de los archivos de esta manera permite un tratamiento flexible de la información, ya que al "correr" el programa estos archivos son igualmente utilizados tanto si se tratan las valoraciones una a una, o todas simultáneamente.

Los archivos modelo son de tipo "TEXT" (que se escriben directamente por el editor del sistema operativo), mientras que los de datos pueden ser del mismo tipo, o bien de tipo

"ASCII" (American Standard Code for Interchange of Information). Estos últimos son los generados por el programa AUTO de adquisición de datos, y pueden ser utilizados tanto para aplicaciones desde BASIC o PASCAL. Los dos tipos de archivos de datos (TEXT y ASCII) son leídos directamente por el programa MINIGLASS. De esta manera, utilizando el programa automático de adquisición de datos, éstos pueden ser tratados inmediatamente después de acabar la valoración por el programa MINIGLASS sin ninguna modificación.

Ejemplos de los archivos 'modelo' y de datos se verán más adelante.

Después de esta breve introducción explicativa, a continuación se tratan algunas de las subrutinas principales utilizadas en este programa. Posteriormente se verá su aplicación y listado completo.

II.2. Descripción del programa MINIGLASS (Subrutinas de control)

II.2.1. Programa principal

Esta parte sólo contiene las rutinas de entrada de datos desde los ficheros y de identificar los tipos de valoración (Gran, Pkas, Complex). Después deja el control al menú principal hasta que el indicador de fin de programa es activado por el usuario.

II.2.2. Menú

Este apartado controla el programa de manera interactiva. Está realizado de forma que después de ejecutar cualquier tarea solicitada por el usuario siempre retorne a esta subrutina hasta que se decida finalizar el programa.

Para que el uso del programa sea sencillo, la asignación de cada tarea sólo precisa del toque de una tecla específica que siempre son recordadas en la pantalla. El esquema de presentación del menú de pantalla es el siguiente:

```
Ajustar_paramet. seLector(n) canviar_reFinament proVar_especies
Refinar Canviar_punts Grafics imPrimir veure_reSultats fI
```

De manera que el usuario sabe qué tecla debe tocar para

realizar un determinado trabajo. Las subrutinas (procedimientos) a las que corresponde la ejecución del programa por el toque de cada letra mayúscula son las siguientes:

A: manual

F: changerefine

V: morespecies

R: refine

C: changepoints

G: grafic

P: printout

S: print

I: pone en funcionamiento el indicador de fin de programa

II.2.3. Manual

Este procedimiento es empleado cuando el usuario desea alterar el valor de cualquier parámetro de los que definen la curva de valoración (concentraciones, potenciales normales, constantes de estabilidad, ...).

Se ha previsto principalmente para acentuar el carácter interactivo del programa. Un caso particular en el que es recomendable su uso es, por ejemplo, cuando en un equilibrio de formación de complejos se postula una nueva especie, se puede cambiar el valor de su constante de formación para ver qué efecto tiene en las curvas de valoración antes de proceder al refinamiento de dicha constante. Además, en los casos en los

que después del refine de determinados parámetros se llegue a un valor de un determinado parámetro que al usuario no le parezca razonable, como puede ocurrir cuando se refinan simultáneamente parámetros no linealmente independientes, a partir de este procedimiento es muy sencillo volver a cambiar a los parámetros de partida. De esta manera no hay por qué parar el programa, sino que se puede continuar normalmente su ejecución.

II.2.4. Changerefine

Sirve para controlar la variable a minimizar cuando trabajamos según el modo Complex. En esta subrutina se puede asignar al indicador 'flag5' el valor de 'V' o 'H', según se desee refinar por volumen o por $p(H)$, respectivamente.

II.2.5. Morespecies

Al igual que la anterior, esta subrutina sólo funciona cuando se trabaja en el modo Complex. Su función principal es para que el usuario pueda añadir o quitar especies al "modelo químico".

II.2.6. Refine

Esta subrutina va a ser comentada mucho más ampliamente, ya que es la encargada de seleccionar y refinar los parámetros

que decida el usuario.

Una de las características destacables en este programa es que, en principio cualquier parámetro que puede afectar a la curva de valoración puede ser refinado (con alguna excepción, como por ejemplo el volumen inicial de la disolución a valorar), y que en cualquier momento el usuario determina qué parámetros van a ser constantes o variables antes de empezar el refinamiento. Los parámetros que son susceptibles de ser refinados pueden variar según el modo en que se trabaje, y son los siguientes:

- Gran: pK_w , J_h , J_{oh} , concentración inicial de ácido o base, E_0 , g y concentración de valorante.
- Pk_{as} : constantes de formación, concentraciones iniciales de ácido y ligando, E_0 , g y concentración de valorante.
- Complex: constantes de formación, concentraciones iniciales en la cubeta de valoración de cualquier componente, concentración de valorante, E_0 y g .

Las únicas restricciones en cuanto al diseño de la experimentación para poder usar este programa y aprovechar todas sus posibilidades de refinamiento de parámetros son, que el valorante debe ser ácido ó base, y añadido con una única bureta. De hecho, éste es uno de los sistemas más utilizados, aunque al diseñar una experimentación puede ser conveniente el

tener más de una bureta, con el objeto de realizar valoraciones a concentración total de alguno de los componentes constante, como hacer valoraciones por retroceso, etc. De todas maneras, el hecho de usar una sola bureta con un componente es una forma de tener menos error experimental, si bien a las rutinas encargadas de resolver los balances de masa les es indiferente un sistema u otro.

En cuanto al funcionamiento de esta subrutina, en primer lugar se inicia preguntando al usuario cuántos parámetros desea refinar simultáneamente (como máximo pueden ser 9), y después le pregunta cuáles son. Estos son introducidos según un número de serie que aparece en la pantalla como recordatorio. El programa selecciona los parámetros a través de la subrutina "Changeparam" comentada posteriormente.

El refinamiento de los parámetros se efectúa según el método iterativo de Gauss-Newton (8, 9, 10) que es utilizado en sus diversas variantes en programas como MINIQUAD (3), SCOGS (8), MINIPOT (5) ó LEAST (11). En nuestro caso se supone que el peso estadístico asignado a cada punto experimental es el mismo, ya que partimos de la base de que la varianza de cada punto en $p(H)$ o en volumen sea la misma. De hecho, esta suposición es totalmente cierta en el caso de que el refinamiento se efectúe según el volumen de valorante (ver ref. 2), y cuando se refina por $p(H)$ se puede suponer que la incertidumbre en su medida es constante en todos los puntos,

especialmente en zonas tamponadas.

A continuación se describe el método de mínimos cuadrados utilizado:

Sea f_i^O la variable observada experimentalmente (es decir, el p(H) o el volumen según el modo de refine) en el punto 'i', que depende según una expresión matemática de una serie de parámetros P_1, \dots, P_m tales como las constantes de formación, Eó, etc. Según el modelo propuesto, a la variable calculada para cada punto experimental le llamaremos f_i^C , tal que esta variable calculada depende de los parámetros según una función: $f^C = \phi(P_1, \dots, P_m)$.

El error cometido en cada punto (residuo o residual) vendrá dado por la diferencia entre el valor experimental y el calculado para la variable observable, es decir:

$$e_i = f_i^O - f_i^C$$

El conjunto de parámetros que mejor ajuste nuestro modelo teórico a los resultados experimentales deberá ser tal que la suma de los cuadrados de los residuales para cada punto sea un valor mínimo (mínimos cuadrados). A esta suma le llamaremos 'U', y por tanto vendrá definida por la ecuación:

$$U = \sum_{i=1}^n (f_i^O - f_i^C)^2$$

-1-

donde 'n' representa el número total de puntos experimentales. En notación matricial se parte de un vector de residuales n-dimensional \vec{e} , tal que el elemento e_i es la diferencia entre la variable observada y la calculada en el punto 'i'. De

esta manera, la función 'U' se puede expresar como:

$$U = \tilde{e}.e \quad (\tilde{e}: e \text{ transpuesto}) \quad -2-$$

El vector gradiente de la función U respecto al conjunto de parámetros nos indica la variación de ésta (y la dirección) al variar los parámetros, y cada elemento de este vector vendrá definido por la derivada parcial de U respecto de cada parámetro:

$$g_j = dU/dP_j \quad -3-$$

$$\text{o bien: } g_j = \sum_{i=1}^n (2 \cdot e_i \cdot de_i/dP_j) = -2 \cdot \sum_{i=1}^n (e_i \cdot df_i^C/dp_j) \quad -4-$$

Teniendo en cuenta que se trabaja en un espacio vectorial de 'm' dimensiones (número de parámetros), se puede asumir que la representación de la función 'U' frente a 'm' ejes de coordenadas es asimilable a la figura de un paraboloides por ser una función cuadrática. En el punto mínimo de esta función 'U' se deberá cumplir que el vector gradiente sea igual a cero (condición de mínimo), es decir que para cada $g_j=0$.

En el caso de que la variable observable sea una función lineal del conjunto de parámetros se podría resolver el conjunto de ecuaciones simultáneamente para obtener directamente el valor del mejor conjunto de parámetros que definan el sistema, sin embargo, en el caso que nos ocupa la relación entre los parámetros y las variables no es lineal, y por tanto la

resolución del sistema (como veremos seguidamente) nos da solamente una mejor aproximación de los parámetros. Es por esto que se debe seguir un proceso iterativo, hasta que los valores finales de los parámetros no puedan ser mejorados. El empleo de esta técnica requiere por tanto, tener "a priori" una primera aproximación de dichos parámetros (no son "calculados" sino "refinados"). El truco utilizado para el caso en que las ecuaciones no son lineales consiste en "linealizarlas" a través de una expansión según las series de Taylor de primer orden (es decir, troncada en el primer término). Sea \vec{P}_0 el vector de parámetros que define el mínimo de la función U , y \vec{P} otro conjunto de parámetros en la vecindad del mínimo. A ambos vectores de parámetros les corresponden unos gradientes \vec{g}_0 y \vec{g} , tal que si se hace una expansión según Taylor se tiene:

$$\Delta g_j = \sum_{k=1}^m (dg_j/dP_k) \cdot \Delta P_k \quad -5-$$

siendo $\Delta g_j = g_{0,j} - g_j$ y $\Delta P_k = P_{0,k} - P_k$

Recordando que por la propia definición de mínimo, el valor de $g_{0,j}$ debe ser igual a cero, la ecuación 5 queda como:

$$\sum_{k=1}^m (dg_j/dP_k) \cdot \Delta P_k = -g_j \quad -6-$$

de manera que los incrementos de los parámetros se pueden encontrar mediante la resolución de este sistema de ecuaciones

lineales. Teniendo en cuenta las ecuaciones 4 y 6 se obtiene:

$$2. \sum_{i=1}^n \langle e_i \cdot df_i^C / dP_j \rangle = \sum_{k=1}^m \langle dg_j / dP_k \rangle \cdot \Delta P_k \quad -7-$$

Poniendo esta última ecuación en forma matricial, se tiene:

$$2. [\tilde{A}] \cdot \vec{e} = [H] \cdot \vec{s} \quad -8-$$

donde $[A]$ representa la matriz de derivadas parciales df_i^C/dP_j de dimensión $n \times m$ (n puntos y m parámetros), $[H]$ la matriz de derivadas parciales del vector gradiente respecto de los parámetros (por tanto, de dimensión $m \times m$) y \vec{s} es el vector de los incrementos en los parámetros. Si nos fijamos en los elementos de la matriz 'H', vemos que éstos son de la forma:

$$h(j,k) = dg_j / dP_k = - 2. \sum_{i=1}^n \frac{d \langle e_i \cdot df_i^C / dP_j \rangle}{dP_k} =$$

$$= 2. \sum_{i=1}^n \left[\langle df_i^C / dP_j \rangle \cdot \langle df_i^C / dP_k \rangle - e_i \cdot \langle d^2 f_i^C / (dP_j \cdot dP_k) \rangle \right] \quad -9-$$

Definimos ahora una nueva matriz $[C]$, de dimensión $m \times m$, tal que sus elementos sean igual a:

$$c(j,k) = \sum_{i=1}^n e_i \cdot \langle d^2 f_i^C / (dP_j \cdot dP_k) \rangle \quad -10-$$

Pasando nuevamente a notación matricial, se ve claramente que:

$$[H] = 2. \langle [\tilde{A}] \cdot [A] - [C] \rangle \quad -11-$$

$$\text{y por tanto: } \langle [\tilde{A}] \cdot [A] - [C] \rangle \cdot \vec{s} = [\tilde{A}] \cdot \vec{e} \quad -12-$$

Finalmente los incrementos de los parámetros se obtienen por:

$$\vec{s} = |B|^{-1} \cdot [\tilde{A}] \cdot \vec{e} \quad -13-$$

Donde la matriz 'B' es igual a: $|B| = [\tilde{A}] \cdot |A| - |C|$

El resultado final de este desarrollo matemático da la ecuación general que se sigue en el método de mínimos cuadrados para el caso en que no se emplee ponderación en los datos. La ecuación 13 representa en realidad el método de Newton-Raphson, mientras que si la matriz 'B' la hacemos igual a: $|B| = [\tilde{A}] \cdot |A|$, se tiene el método de Gauss-Newton. La diferencia entre ambos métodos reside en la inclusión o no de la matriz C definida en la ecuación 10. A primera vista, es obvio que el método de Newton-Raphson conteniendo las derivadas de segundo orden es más riguroso que el de Gauss-Newton y que debería ser más rápido en la busca del mínimo de una función. Sin embargo, algunos investigadores han demostrado experimentalmente (8, 11) que un problema de este tipo es resuelto más rápidamente por el método de Gauss-Newton. Este hecho podría explicarse por la necesidad de un número mayor de evaluaciones de una función debido a la necesidad de encontrar las segundas derivadas, aunque de hecho parece ser además que el número de ciclos necesarios para llegar al mínimo en una función es menor en el método de Gauss-Newton.

La obtención de la matriz 'B' nos permite además una estimación de las varianzas de los parámetros, una vez llegado al mínimo en la función U (ver ref. 4). La matriz varianza-covarianza de los parámetros está definida por:

$$|M| = \frac{U_{\min}}{n - m} \cdot |B|^{-1}$$

-14-

Los elementos diagonales de esta matriz son las varianzas estimadas de los parámetros, mientras que los otros elementos representan la covarianza entre ellos. De hecho, durante el desarrollo del programa sólo se calculan las varianzas por ser éstas de mayor interés.

En nuestro programa, como se ha indicado anteriormente, las derivadas necesarias según el desarrollo previo son obtenidas numéricamente, debido a la gran variedad de parámetros ajustables que existen y a la posibilidad de efectuar el refine según dos modos (Vt y p(H)). Por su parte, el empleo de las derivadas numéricas ha sido cuestionado en cuanto a su lentitud (ya que se deben realizar más evaluaciones de la función U) y a que por no ser exactas, se pierde precisión en los sucesivos pasos (6, 12) por ser sólo una aproximación. El primer problema no es muy grave, ya que según nuestra experiencia se obtienen resultados razonables en cuanto al tiempo de cálculo, mientras que el segundo puede tener importancia en los casos en los que en los sucesivos ciclos de refine la función U diverja, lo que normalmente puede llegar a ocurrir cuando al tomar incrementos para efectuar la diferenciación numérica se tomen de una magnitud errónea o también, a que en realidad si se tienen unos parámetros de partida no demasiado buenos el proceso de refine pueda ir totalmente equivocado.

En nuestro caso, estos problemas de divergencia se han intentado evitar en lo posible mediante el uso de un parámetro que acorta o alarga el vector de los incrementos de manera que le obligue a converger ("convergence forcer"). Se ha empleado el método propuesto por Wentworth (13, 14), muy sencillo en teoría y fácil de implementar, obteniéndose resultados totalmente satisfactorios. Este método consiste en suponer que existe un cierto escalar λ , tal que multiplicando éste por el vector de corrección \vec{s} se obtiene una reducción efectiva en la función U . Un procedimiento sencillo es asumir que U es una función cuadrática de λ (forma de parábola) según:

$$U = a + b.\lambda + c.\lambda^2 \quad -15-$$

Sea λ_{\min} el valor de este escalar que corresponde a un mínimo U_{\min} . Por ser condición de mínimo se debe cumplir:

$$dU/d\lambda = 0 = b + 2.c.\lambda_{\min} \quad -16-$$

$$\text{de donde: } \lambda_{\min} = -b/2.c \quad -17-$$

La resolución de la ecuación 17 se realiza hallando los valores de 'a', 'b' y 'c' a través de los valores de U en los casos en los que λ sea igual, respectivamente, a 0, 0.5 y 1:

$$\text{- si } \lambda = 0 : U_0 = a \quad -18-$$

$$\text{- si } \lambda = 0.5 : U_{0.5} = a + 0,5.b + 0,25.c \quad -19-$$

$$\text{- si } \lambda = 1 : U_1 = a + b + c \quad -20-$$

Combinando las ecuaciones 19 y 20 despejamos:

$$c = 2.U_1 + 2.U_0 - 4.U_{0.5} \quad -21-$$

$$y \quad b = U_1 - U_0 - c \quad -22-$$

Sustituyendo finalmente los valores de 'b y 'c' en la ecuación 17 se obtiene:

$$\lambda_{\min} = \frac{1}{2} + \frac{U_0 - U_1}{4.(U_0 + U_1 - 2.U_{0.5})} \quad -23-$$

Este es el procedimiento seguido para la optimización del vector de los incrementos en los parámetros, de manera que se asegura la convergencia en el proceso iterativo. En nuestro programa, este sistema está diseñado de manera que normalmente se sigue el método de Gauss-Newton desarrollado anteriormente, pero en el momento en que la función U comienza a diverger, se utiliza este método de optimización de los parámetros ; si después de esto se consigue una reducción en U , se vuelve otra vez al método de Gauss-Newton normal, y en caso contrario (que no disminuya U) el proceso de refine es interrumpido y en la pantalla del ordenador aparece un mensaje de atención. De todas maneras, la subrutina de refine está "asegurada" de manera que siempre retiene los parámetros que corresponden a un mínimo de U.

Por otra parte, esta subrutina incluye un controlador que impide que los incrementos aplicados a los logaritmos de las constantes de formación sean mayores que 2 unidades.

II.2.7. Changepoints

Esta subrutina sirve para escoger los puntos experimentales que están siendo considerados en los cálculos, de forma similar a la empleada en el programa MINIPOT (5) . La principal utilidad puede ser, para realizar un calibrado de electrodos (refinar la Eó) en la zona inicial de una valoración en la que aún no existe complejación ó es despreciable, ó para escoger zonas determinadas de pH en donde predomine una determinada especie, etc.

Se ejecuta siempre al empezar a funcionar el programa para preguntarle al usuario qué puntos quiere incluir de cada valoración; si durante el funcionamiento del programa éstos se desean volver a cambiar se llama a esta subrutina desde el menú principal.

II.2.8. Grafic

Contiene principalmente el menú auxiliar para el sistema de gráficos, cuyas funciones son para preguntar al usuario qué tipo de gráficos quiere realizar (curvas de valoración o distribución de residuales) y si van a ser enviados a la pantalla o al "plotter".

Por otra parte tiene en cuenta los límites superior e inferior de pH y los volúmenes de valorante para ajustarlos automáticamente a un tamaño de dibujo conveniente.

II.2.9. Print y Printout

Estas subrutinas son las encargadas de representar los resultados numéricos obtenidos. La primera los dirige a la pantalla del ordenador, mientras que la segunda está enfocada para la impresora.

La diferencia básica entre ellas es que "Print" realiza los caracteres de la pantalla con sistemas de video inverso, medio brillo, etc. de manera que su presentación sea más clara y agradable para el usuario.

II.2.10. Selector

No es una subrutina como el caso de las anteriores, sino que se trata de una variable lógica utilizada para tener en cuenta o no el selector de especies. En caso de que la variable "selector" sea cierta, una especie es rechazada cuando su desviación estándar estimada es mayor que la mitad del valor absoluto de la constante (en unidades logarítmicas).

El valor de "verdadero" o "falso" a esta variable se asigna directamente desde el menú cuando se trabaja en el procedimiento Complex.

II.3. Subrutinas de cálculo de balances de carga y masa

Tienen como función calcular las variables observables teóricas a partir de los datos iniciales de volúmenes, concentraciones, constantes de formación, etc. Como hemos indicado anteriormente, en los modos de cálculo Gran y Pkas se calcula la concentración de ión hidrógeno libre teórica, mientras que en el modo Complex puede ser ésta la función calculada o bien el volumen de valorante añadido.

II.3.1. Gran

La concentración de hidrógeno libre teórica es calculada en este caso a partir del balance de cargas en la disolución valorada.

Si consideramos la valoración de una disolución de ácido fuerte (HCl, por ejemplo) con una base fuerte como el hidróxido sódico, se cumplirá en todo momento que:



Sea ' H_0 ' la concentración inicial de una disolución de HCl de volumen inicial ' V_0 ', y ' V_t ' el volumen de NaOH añadido de concentración ' H_t '. Si llamamos ' h ' a la concentración de ión hidrógeno libre y sustituimos el valor de $[OH^-]$ por K_w/h (siendo K_w la constante de autoprotólisis del disolvente), la ecuación 24 se transforma en:

$$h + \frac{V_t \cdot H_t}{V_0 + V_t} = \frac{H_0 \cdot V_0}{V_0 + V_t} + \frac{K_w}{h} \quad -25-$$

Haciendo el cambio: $V_{tot} = V_o + V_t$ y multiplicando todos los términos por 'h' se obtiene la ecuación cuadrática:

$$h^2 + \frac{h \cdot (H_t \cdot V_t - H_o \cdot V_o)}{V_{tot}} - Kw = 0 \quad -26-$$

Y finalmente:

$$h = \frac{(H_o \cdot V_o - H_t \cdot V_t) / V_{tot} + \left(\left((H_t \cdot V_t - H_o \cdot V_o) / V_{tot} \right)^2 + 4 \cdot Kw \right)^{1/2}}{2} \quad -27-$$

II.3.2. Pkas

Para el cálculo de 'h' se emplea en esta subrutina un sofisticado algoritmo propuesto por Martell y Motekaitis en su programa PKAS (15). Presenta la particularidad de haber sido diseñado específicamente para el estudio de equilibrios de disociación ácido-base y tiene como mayor ventaja, al ser comparados con algoritmos de tipo general (como por ejemplo el descrito posteriormente en el procedimiento Complex), el de ser apreciablemente más rápido. Esta mayor rapidez es debida a que un algoritmo general estaría basado en la búsqueda de dos variables (la concentración libre del anión y la de H^+), mientras que el algoritmo de PKAS sólo busca una única variable (la concentración de ión hidrógeno libre).

El procedimiento para calcular la concentración teórica libre del ión hidrógeno está basado en el método iterativo de Newton (16) o también llamado método de las tangentes. El fundamento matemático se describe brevemente a continuación para una mejor comprensión del algoritmo usado:

Supongamos una ecuación ' $f(x)$ ' que depende de una variable independiente ' x ', y que existe una raíz de esta ecuación θ , tal que: $f(\theta) = 0$. Si tenemos una aproximación ' x_n ' lo suficientemente cercana a θ , se puede mejorar según el método de Newton por la relación:

$$\Delta x_n = \theta - x_n \quad -28-$$

Mediante las series de Taylor el valor de $f(\theta)$ viene dado por:

$$f(\theta) = 0 = f(x_n + \Delta x_n) = f(x_n) + \Delta x_n \cdot f'(x_n) \quad -29-$$

siempre y cuando Δx_n sea un valor muy pequeño. Despejando la ecuación 29 podemos deducir el valor de Δx_n :

$$\Delta x_n = - f(x_n) / f'(x_n) \quad -30-$$

La nueva aproximación de la variable ' x ' vendrá dada por:

$$x_{n+1} = x_n + \Delta x_n = x_n - f(x_n) / f'(x_n) \quad -31-$$

Este proceso iterativo se va repitiendo hasta que el valor de $f(x_n)$ llega a tener un valor lo suficientemente pequeño (según las necesidades de cada caso), y poder considerar que el valor de x_n será una estimación conveniente de la raíz real θ .

El empleo del método de Newton implica el uso de una función diferenciable que tenga un valor igual a cero en el momento de hallar la raíz exacta de la ecuación propuesta.

En el algoritmo de Martell y Motekaitis, la función empleada está representada por:

$$F(h) = 0 = \frac{\sum_{i=1}^n \beta_i \cdot i \cdot h^i}{1 + \sum_{i=1}^n \beta_i \cdot h^i} - \frac{H_{\text{tot}} - h + Kw/h}{A_{\text{tot}}} \quad -32-$$

Donde h es la concentración de hidrógeno libre buscada, H_{tot} la concentración total de hidrógeno disociable, A_{tot} la de ligando, y β_i las sucesivas constantes de formación de las especies protonadas. Este método supone definir una función $F(h)$ que es igual a cero en el momento en que la \bar{j} teórica según las constantes de formación sea igual a la \bar{j} que correspondería para las condiciones de H_{tot} y A_{tot} a una concentración de ión hidrógeno libre común para las dos. Este valor de h es buscado iterativamente según el método de Newton de manera que en la iteración $n+1$ será igual a:

$$h_{n+1} = h_n - F(h_n)/F'(h_n) \quad -33-$$

siendo "F'" la derivada de la función F respecto de h.

La implementación de este sistema en nuestro programa presenta una variación respecto del original, que reside en que las derivadas no se toman respecto a 'h', sino respecto a 'ln(h)'. Se ha procedido de esta manera ya que, como veremos, nunca puede dar un resultado negativo de 'h'. En el sistema propuesto, las iteraciones vienen dadas por las ecuaciones:

$$\ln(h_{n+1}) = \ln(h_n) - F(h_n)/F'(h_n) \quad -34-$$

(en este caso F' representa la derivada de F respecto ln(h))

$$\begin{aligned} \ln(h_{n+1}) &= \ln(h_n) - \ln(\exp(F(h_n)/F'(h_n))) = \\ &= \ln(h_n/\exp(F(h_n)/F'(h_n))) \end{aligned} \quad -35-$$

y finalmente, tomando antilogaritmos se obtiene:

$$h_{n+1} = \frac{h_n}{\exp(F(h_n)/F'(h_n))} \quad -36-$$

Sólo es necesario conocer la derivada de F respecto de ln(h), cuya obtención es inmediata: sea $d(\ln(h)) = d(h)/h$, de donde : $d(F)/d(\ln(h)) = h \cdot d(F)/d(h)$. Por tanto su deducción es sencilla según:

$$\frac{d(F)}{d(\ln(h))} = - \frac{(h+kw/h)}{A_{tot}} - \frac{\sum_{i=1}^n i^2 \cdot \beta_i \cdot h^i}{1 + \sum_{i=1}^n \beta_i \cdot h^i} - \frac{(\sum_{i=1}^n i \cdot \beta_i \cdot h^i)^2}{(1 + \sum_{i=1}^n \beta_i \cdot h^i)^2} \quad -37-$$

Como comentario adicional respecto al funcionamiento de esta subrutina, diremos que se toma como aproximación inicial de 'h' su valor experimental, y el proceso se repite iterativamente hasta que el valor de $F(h)$ llega a ser menor que 1×10^{-4} , suficientemente pequeño ya que representa un error del orden de una magnitud inferior al error experimental.

II.3.3. Complex

Una de las innovaciones presentadas en el programa reside en esta subrutina, que es la encargada de resolver los balances de materia de los componentes del sistema. El método de resolución de las ecuaciones es totalmente clásico, mediante el método iterativo de Newton descrito en el apartado anterior, pero adaptado para la resolución simultánea de un sistema de varias ecuaciones no lineales (16).

Su característica diferenciadora reside en que la misma subrutina sirve para calcular el valor teórico de 'h' si suponemos el volumen añadido como variable independiente, y viceversa. Esto se consigue con dos premisas: que sólo haya una bureta de valorante (puede ser ácido ó base), y que si definimos un sistema en equilibrio con 'n' componentes, el ión hidrógeno siempre será nombrado como el 'n-simo' componente. La ordenación del resto de los componentes del sistema es totalmente indiferente, aunque debe ser explicitada en el archivo "modelo".

A continuación se muestra el fundamento matemático para el caso en el que el refinamiento se efectúe según $p(H)$, ya que ambos modos son totalmente similares en cuanto a cálculo.

Supongamos que tenemos un sistema en equilibrio de 'n' componentes y que pueden formar hasta 'k' especies. Las concentraciones libres de cada componente están asignadas al vector ' \vec{cl} ', de manera que ' $cl(i)$ ' representa la concentración libre en el equilibrio del componente 'i'. Definimos además una matriz de coeficientes estequiométricos ' cf ' de dimensión $k \times n$, según lo cual el coeficiente del componente 'i' en la especie 'j' será igual a ' $cf(j,i)$ '.

Si β_j es la constante estequiométrica de formación (es decir, en términos de concentración) de la especie 'j', la concentración de ésta en el equilibrio será igual a:

$$S_j = \beta_j \cdot \prod_{i=1}^n cl(i)^{cf(j,i)} \quad -38-$$

Consideremos ahora un vector ' \vec{G} ' que contiene los balances de masa teóricos de todos los componentes. Sus elementos vendrán definidos por:

$$G(i) = cl(i) + \sum_{j=1}^k cf(j,i) \cdot S_j \quad -39-$$

Las ecuaciones 38 y 39 tienen dos tipos de variables: las constantes de formación de las especies complejas (y sus coeficientes) que son conocidas ó por lo menos una estimación de

las que pretendemos refinar, y las concentraciones libres de los componentes que en principio son desconocidas. Para poder plantear las ecuaciones de balances de masa teóricos necesitamos saber, pues, los valores de las concentraciones libres. Esto define un sistema de 'n' ecuaciones no lineales (el vector G) y 'n' incógnitas (el vector cf) que se puede resolver por el método de Newton. Para ello se define otro vector \vec{F} como la diferencia entre las concentraciones totales de cada componente experimentales (T(i)) y las calculadas según la ecuación 31, es decir:

$$F(i) = G(i) - T(i) \quad -40-$$

Esta última ecuación indica claramente que los términos del vector 'F' son iguales a cero cuando las concentraciones libres calculadas iterativamente estén de acuerdo a la vez con el " modelo químico" propuesto y las concentraciones totales que tenemos en realidad.

El desarrollo del método de Newton para un sistema de varias ecuaciones es totalmente similar al caso anterior en el que se tenía una sola ecuación con una incógnita, pero ahora la derivada del vector \vec{F} respecto de \vec{c}_l nos dará una matriz (Jacobiana) en lugar de una sola expresión. También como en el caso anterior, y como indican otros autores (17), las derivadas se toman respecto a los logaritmos neperianos de las concentraciones libres para evitar que éstas (o las concentraciones de las especies complejas) puedan llegar a dar resultados

negativos. Siendo \vec{L} el vector de los logaritmos de las concentraciones libres, los elementos de la matriz jacobiana serán de la forma:

$$J(i, l) = \frac{d(F(i))}{d(L(l))} = \delta_{il} \cdot cl(i) + \sum_{j=1}^k cf(j, i) \cdot cf(j, l) \cdot S_j \quad -41-$$

donde el término δ_{il} representa la "delta de Kronecker", que es igual a la unidad si $i=l$, y cero en caso contrario.

Durante el proceso iterativo, los valores del vector \vec{L} en la iteración 'm+1' vendrán dados por:

$$\vec{L}_{m+1} = \vec{L}_m - \vec{F}_m \cdot |J|^{-1} \quad -43-$$

Si se define un nuevo vector \vec{R} , tal que $\vec{R} = \vec{F} \cdot |J|^{-1}$, y se hace el cambio en la ecuación 43, se obtiene:

$$\vec{L}_{m+1} = \vec{L}_m - \vec{R}_m \quad -44-$$

Los elementos particulares del vector \vec{L} se pueden escribir como:

$$L(i)_{m+1} = \ln(cl(i)_{m+1}) = \ln(cl(i)_m) - \ln(\exp(R(i)_m)) \quad -45-$$

y en consecuencia:

$$cl(i)_{m+1} = cl(i)_m / \exp(R(i)_m) \quad -46-$$

El ciclo iterativo se repite hasta que la suma de los valores absolutos de los elementos del vector \vec{F} es menor que 10^{-9} ; si se tiene en cuenta que normalmente las concentraciones totales de los componentes son en realidad del orden entre

0.01 - 0.0001 M, representa que el error cometido en estos cálculos es inferior a una parte por millón.

Una vez resuelto este sistema de ecuaciones, el residual (D) se define en cada punto como la diferencia entre el p(H) experimental y el calculado. Teniendo en cuenta que el hidrógeno es el n-simo componente, se tiene:

$$D = p(H)_{\text{exp}} - p(H)_{\text{calc}} = p(H)_{\text{exp}} + \log(\text{cl}(n)) \quad -47-$$

En el caso en que el usuario selecciona el modo de refine por volumen de valorante, los cálculos e iteraciones anteriores son efectuados de manera totalmente similar, con la diferencia de que ahora el sistema de ecuaciones se resuelve para 'n-1' componentes, ya que se toma como fijo el valor experimental de H^+ (en nuestra notación como cl(n)). Una vez resueltas las n-1 ecuaciones, y conocidas las concentraciones en equilibrio de todas la especies, veamos el balance de masas para el ión hidrógeno. Sea 'Th₀' la concentración analítica total inicial de hidrógeno dissociable, y 'Th' en cualquier punto de la valoración. Si consideramos que la concentración de valorante 'H_t' puede ser positiva (si valoramos con ácido) o negativa (con base), el balance de masa vendrá dado en cualquier punto por:

$$Th = \frac{Th_0 \cdot V_0 + H_t \cdot V_t}{V_0 + V_t} = H^+ + \sum_{j=1}^k \text{cf}(j,n) \cdot S_j \quad -48-$$

El término de la derecha de esta última ecuación es equivalente al elemento n del vector \vec{G} definido en la ecuación 39. Reordenando la ecuación 48 se obtiene:

$$Th_o \cdot V_o + H_t \cdot V_t = G(n) \cdot (V_o + V_t) \quad -49-$$

y despejando V_t :

$$V_t = \frac{(Th_o - G(n)) \cdot V_o}{G(n) - H_t} \quad -50-$$

Resultando de esta manera el volumen teórico ($V_{t,calc}$) de valorante añadido. El residual es en este caso:

$$D = V_{t,exp} - V_{t,calc} \quad -51-$$

Siguiendo este esquema se puede realizar el cálculo teórico de $p(H)$ ó V_t con muy pocas diferencias en su desarrollo. Comparando ambos modos se observa que el vector \vec{G} es actualizado en cada iteración ya sea por $p(H)$ ó V_t , mientras que la diferencia fundamental (aparte del cálculo final de V_t , que es inmediato), es que el vector de concentraciones libres se varía en su totalidad en el modo por $p(H)$, mientras que en el modo por V_t , el último elemento de ese vector (que es H^+) se mantiene fijo. Esto hace que refinando por volumen de valorante el cálculo sea comparativamente más sencillo, ya que el sistema de ecuaciones es resuelto para una variable menos. A efectos de tiempo de cálculo no se han encontrado diferencias apreciables, excepto en zonas de pH no tamponadas en las cua-

les la resolución del sistema de ecuaciones puede ser más problemática y tardar más; de hecho en las zonas no tamponadas o que pertenezcan a puntos finales de valoración (donde los errores de concentración ya pueden llegar a ser muy importantes) es más recomendable emplear el modo de refine según Vt ya que aparte de que la solución sea más segura, es evidente que las varianzas en volumen serán en este caso mucho más homogéneas que en p(H).

Esta subrutina incorpora, por otra parte, un controlador que impide que la variación en una concentración libre de una iteración a otra sea mayor que 10 veces su valor. En ese caso, se restringe su cambio a uno máximo de 5 veces, aunque manteniendo la misma dirección (de aumento o disminución).

Otra característica de esta subrutina es que es totalmente general, ya que por su diseño puede trabajar para cualquier sistema siempre y cuando el número de componentes sea mayor que uno. Sin embargo, el programa la limita en principio a cuatro componentes que pueden ser aumentados simplemente re-dimensionando las matrices utilizadas. Además no distingue entre metal o ligando sino que, como se ha comentado previamente, la única restricción por esta parte es que el último componente sea el ión hidrógeno.

II.4. Subrutinas auxiliares

En este apartado serán comentadas brevemente algunas subrutinas que, si bien son totalmente "transparentes" al usuario, son necesarias para el funcionamiento del programa.

II.4.1. Changeph

Se encarga de calcular el valor de $p(H)$ experimental en cada punto de la valoración a partir de los datos de E_0 , g , potenciales de unión líquida ácida y básica y del potencial medido (E) a partir de la ecuación:

$$E = E_0 + g \cdot \log([H^+]) + J_h \cdot [H^+] + J_{oh} \cdot Kw/[H^+] \quad -52-$$

Debido a que es una ecuación no lineal, para su resolución se emplea nuevamente el método iterativo de Newton descrito en el apartado II.3.2.

Aunque otros autores (ver SUPERQUAD, ref. 18) no consideran el valor de la unión líquida en los cálculos de este tipo por considerarla despreciable en la mayoría de los casos, en nuestro caso se ha creído conveniente incluirla ya que no implica una mayor complicación en el programa.

II.4.2. Matinv

Esta es una subrutina general de inversión de matrices que emplea el método de eliminación de Gauss-Jordan (19). Su uso viene comandado por los procedimientos Refine y Complex, que en sus algoritmos necesitan de una inversión de matrices.

II.4.3. Changeparam

Aunque sea relativamente corta en extensión, esta subrutina es una parte principal del método de refine empleado por nosotros, ya que al ser variados los parámetros durante el proceso iterativo de refinamiento, éstos deben ser controlados y discriminados por "Changeparam".

Su fundamento consiste en reconocer dos tipos de parámetros: los comunes y los específicos de cada valoración en particular. Considera que cualquier parámetro que tenga un número de serie entre 1 y 20 es un parámetro común (ver procedimiento Refine), como por ejemplo una constante de formación ó un potencial de unión líquida, mientras que si es mayor que 20 indica un parámetro específico de una valoración. El tipo de parámetros varía según qué decena sea asignada a su número de serie, por ejemplo, los parámetros veinti... indican que se trata de un potencial normal de electrodo, treinta... es la pendiente del electrodo (g), y los valores superiores indican concentraciones de cada componente en particular; a su

vez, las unidades nos dicen a qué valoración corresponde. De esta manera, un parámetro etiquetado como "21" quiere decir que equivale al potencial normal de electrodo de la primera valoración, y "32" sería el valor de "g" de la segunda.

Diseñado de esta manera, el programa permite trabajar con hasta 20 constantes de formación de especies complejas que pueden ser ajustadas (separadamente, ya que el número máximo de parámetros que pueden ser refinados simultáneamente es de 9).

Otra función que controla esta subrutina es si al cambiar el valor de un parámetro es necesario continuar la ejecución del programa por otra parte. Por ejemplo, si estamos refinando potenciales normales, además de cambiar el parámetro en sí, se deben recalcular también los valores de $p(H)$ experimental de todos los puntos de la valoración correspondiente.

II.4.4. Setconc

Su función es calcular la concentración total de cada componente en cada punto de la valoración. Se inicia al principio del programa, pero es nuevamente activada cada vez que "Changeparam" detecta que hay un cambio de concentración total de algún componente ó que varía la concentración de valorante debido a un refinamiento o un ajuste manual de los parámetros (en "Changeph" se sigue el mismo esquema).

II.4.5. Titcurve y Distr

Estas subrutinas son las encargadas del ajuste automático de la escala y del dibujo propiamente dicho de los gráficos de curva de valoración y de distribución de residuales, respectivamente.

Los símbolos asignados en cada gráfica a los puntos experimentales son distintos para cada curva, existiendo 9 símbolos en total (por ejemplo, la primera curva es representada por cuadrados, la segunda por triángulos, etc.) que son seleccionados y dibujados por las subrutinas auxiliares 'cuadr', 'circle' y 'plot'. Para la realización de estos diferentes símbolos no se ha recurrido a la librería de gráficos de H.P. sino que son definidos por programa a partir de las instrucciones sencillas "move" y "line" que son idénticas o equivalentes en su función a las usadas por otros ordenadores. Esto obedece al hecho de que se ha intentado restringir al máximo las rutinas específicas de H.P. para preveer su mejor "transportabilidad" a otros tipos de ordenadores.

II.5. Comentarios adicionales

Nuestra experiencia con este programa ha puesto de manifiesto en primer lugar su potencia en cuanto a capacidad de cálculo, que aprovecha en buena parte las características del ordenador HP 9816 S del que se ha dispuesto. Como ejemplo, podemos decir que para el sistema de prueba Ni(II)-glicina que se muestra a continuación, el refinamiento simultáneo de las tres constantes de formación sucesivas para un total de aproximadamente unos 210 puntos experimentales pertenecientes a cinco valoraciones distintas se ha llevado a cabo en cuatro ciclos iterativos en un tiempo de 11-12 minutos. Por supuesto que un programa de cálculo tipo MINIQVAD o LETAGROP usado en un "mainframe" tardaría un tiempo muy inferior para llegar a los mismos resultados, pero no hay que olvidar que, por otra parte, el tiempo tardado es más que aceptable para un microordenador, y además teniendo en cuenta las facilidades de manejo que presenta el enfoque interactivo de este programa. Otro factor a recordar cuando hacemos comparaciones, es que el tener una copia en papel de los resultados, ó trazar un gráfico en pantalla ó plotter, etc., son trabajos que sólo le cuestan al usuario el tocar una tecla y que son obtenidos inmediatamente.

Volviendo nuevamente al punto de comparación de tiempo en efectuar los cálculos, hay que recordar que por otra parte, el

uso de un "mainframe" viene condicionado a su uso en tiempo compartido o ejecución por "batch" (con sus correspondientes pérdidas de tiempo para el usuario), de manera que la diferencia efectiva de tiempo entre un sistema u otro no es tan grande como en principio podría pensarse.

Otro factor a tener en cuenta cuando se compara un programa para ordenador "grande", es que el programa usado ha sido desarrollado conjuntamente con el sistema de adquisición automática de datos descrito en el apéndice anterior, de manera que se puede dar el caso (totalmente posible) de que el usuario no haya tenido que escribir ningún archivo de datos de las valoraciones, sino sólomente el archivo "modelo". A nuestro parecer, ésta es una de las grandes ventajas que presenta el programa, ya que si consideramos que con la unidad de adquisición de datos ("data logger") HP 3421 A empleada se pueden llegar a realizar hasta 8 valoraciones automáticas simultáneamente (en el caso de que se dispusieran de 8 pH-metros y 8 autoburetas), los cálculos correspondientes se pueden llevar a cabo inmediatamente después de acabadas las valoraciones; si consideramos además el tiempo que puede tardar el usuario en escribir la misma cantidad de datos para ser tratados por MINIQVAD, por ejemplo, la ganancia de tiempo usando el programa descrito es ciertamente notable.

Por otra parte, el uso de gráficos para representar los resultados es una de las características que hacen más senc-

llo el uso de este programa: a simple vista el usuario puede apreciar de una manera intuitiva si hay alguna valoración que necesite un ajuste ó refine en su Eó por ejemplo, ó si la inclusión de una nueva especie en el modelo es realmente necesaria ó por el contrario está de más, etc.

El diseño de este programa totalmente modular hace que sean perfectamente posibles posteriores modificaciones en su funcionamiento. En principio, una de las variaciones posibles que se pueden realizar sin ningún problema es un pequeño cambio para aceptar valoraciones cuyos datos sean pH - Vt además de los usados E - Vt , ya que sólo hay que tener en cuenta el coeficiente de actividad para el medio iónico empleado. Este fácil cambio es posible ya que el programa refina respecto a " $-\log H^+$ ".

Otras posibles modificaciones sencillas de realizar son, aceptar datos de múltiples buretas con cualquier composición de valorante, ó la posibilidad de no trabajar con electrodos de pH sino que se use otro par electródico para seguir la concentración de metal ó ligando libres. Por supuesto, estos últimos cambios ya implican una pérdida en cuanto a la capacidad del programa, pues haría necesario el refine por potencial leído en lugar de p(H), y no se podría refinar respecto a volumen de valorante.

A su vez, el empleo de un método de refine basado en derivadas numéricas también hace posible su posterior uso para

el cálculo de constantes de estabilidad a partir de otros tipos de datos, como espectrofotométricos, entalpimétricos, etc. con relativamente pocos cambios.

Como comentario final digamos que el programa presentado puede ser un ejemplo de las posibilidades de los microordenadores de la última generación (dentro de muy poco ya serán más potentes, más baratos y de menor tamaño) , que con el uso de microprocesadores rápidos y de gran potencia (por ejemplo, el Motorola 68000 empleado por el HP 9816 S ó Intel 80286 en el IBM PC/AT), gran capacidad de memoria (más de 1 MByte) y con sofisticados sistemas operativos (UNIX y sus derivados) y nuevos lenguajes de programación como Modula-2 ó ADA hacen que en la actualidad la frontera entre un miniordenador y un ordenador personal sea cada vez más pequeña (ver ref. 20).

II.6. Estructura de los ficheros

Como se ha comentado previamente, existen dos tipos de ficheros, el del "modelo químico" y el de los datos propiamente dichos. La organización de éstos últimos ya se ha visto al describir el programa de valoración automática, por tanto explicaremos cómo se escriben los archivos "modelo", ya que pueden variar según qué tipo de valoración efectuamos:

- Tipo Gran : contiene sólo los valores de pK_w , J_h y J_{oh} . Para el caso particular del medio en que se ha estudiado la complejación entre el ión Ni^{2+} y la glicina es: 13.71 0 0 . Nótese que aquí no se considera la unión líquida ácida y básica por ser muy pequeñas y que además, como el resto de ficheros que veremos, los números no están separados por comas.
- Tipo Pkas : además de los datos del tipo Gran, contiene el número de protones que aporta cada molécula de ligando, el número de especies protonadas, y el logaritmo de su constante de formación estimada. Por ejemplo: 13.71 0 0 1 2 9.6 12.1 en el caso de la protonación del ión glicinato.

- Tipo Complex : este archivo contiene la información de pKw, Jh y Joh como los anteriores, seguida del número de componentes del sistema, el número de especies complejas, los protones que aporta cada componente, y los coeficientes estequiométricos de cada especie y los logaritmos de sus constantes de formación. En el caso del sistema Ni(II)-glicina, este archivo es de la forma:

```

13.71 0 0 3 5 1 0 1
1 0 1 9.66
1 0 2 12.07
1 1 0 5.62
2 1 0 10.34
3 1 0 13.71

```

El ordenamiento de los componentes es de : ligando-metal-protón, pero no tiene por qué ser necesariamente así. Como se ha dicho anteriormente al comentar la subrutina "Complex", sólo el protón debe ser siempre el último.

II.7. Ejemplo de aplicación : sistema Ni(II)-Glicina

Las razones del empleo de este sistema para comprobar el buen funcionamiento tanto del sistema automático de valoración como del programa desarrollado ya se han comentado en la introducción del apéndice I, por tanto en este apartado se comentarán los resultados obtenidos en su aplicación.

En total se ha efectuado un conjunto de 10 valoraciones potenciométricas repartidas en tres series, que serán nombradas como "G", "P" y "C". Las valoraciones G.1 y G.2 se han utilizado para determinar los parámetros de pK_w y potenciales de unión líquida (éstos han sido encontrados despreciables); las P.1, P.2 y P.3 para la determinación de las constantes de disociación de la glicina, y la serie C.1 a C.5 para el cálculo de las constantes de formación de los complejos.

Con el objeto de que nuestras medidas fueran comparables con las de la bibliografía, todas las valoraciones se han llevado a cabo en un medio iónico constante de 1.00 M en cloruro sódico, y a una temperatura de 25°C.

II.7.1. Experimental

Todas las disoluciones empleadas han sido preparadas a partir de reactivos de calidad P.A. Estos productos han sido:

- glicina (ácido amino-acético) Scharlau.
- NiCl_2 y HCl Carlo Erba.
- NaOH y NaCl Merck.

Las valoraciones potenciométricas se han efectuado según el procedimiento descrito en el capítulo cuarto, manteniendo un burbujeo constante de N_2 (exento de oxidantes) y con agitación magnética. La solución valorante ha sido de hidróxido sódico aproximadamente 0.1 M , y la temperatura se ha mantenido constante a $25 \pm 0.1^\circ\text{C}$.

La estandarización de las disoluciones de los reactivos se ha llevado a cabo mediante los métodos usuales (21): el Ni(II) por gravimetría como dimetilglioximato; las disoluciones de hidróxido sódico y ácido clorhídrico frente a ftalato ácido de potasio y bórax, respectivamente; la pureza de la glicina empleada se ha controlado mediante valoración potenciométrica resultando ser mayor de un 99.9 % . Las disoluciones de cloruro sódico se han preparado por pesada del producto previamente secado a 110°C . Todas las disoluciones han sido preparadas con agua bidestilada recientemente hervida.

Los aparatos utilizados en el estudio de este sistema han sido los siguientes:

- bureta automática Metrohm E 415 , con cilindro de 10 ml de capacidad.

- pH-metro Radiometer, modelo PHM-84.
- electrodos Radiometer, de vidrio G 202 B (de bajo error alcalino) y de referencia Ag/AgCl K 801.
- termostato Selecta mod. Tectron.
- sistema automático de valoración descrito en el apéndice I.

Los datos experimentales obtenidos en las valoraciones potenciométricas se incluyen en el apartado II.10.

II.7.2. Resultados

A continuación se resumen los resultados obtenidos para las distintas series de valoraciones en el estudio de este sistema. En primer lugar las correspondientes al calibrado de los electrodos y caracterización del medio iónico, y posteriormente las de disociación ácida y de formación de complejos.

II.7.2.1. Valoraciones G.1 y G.2

En primer lugar, el tratamiento de estos resultados se llevó a cabo según las funciones de Gran (22). En la figura II.1 están representadas las gráficas correspondientes a

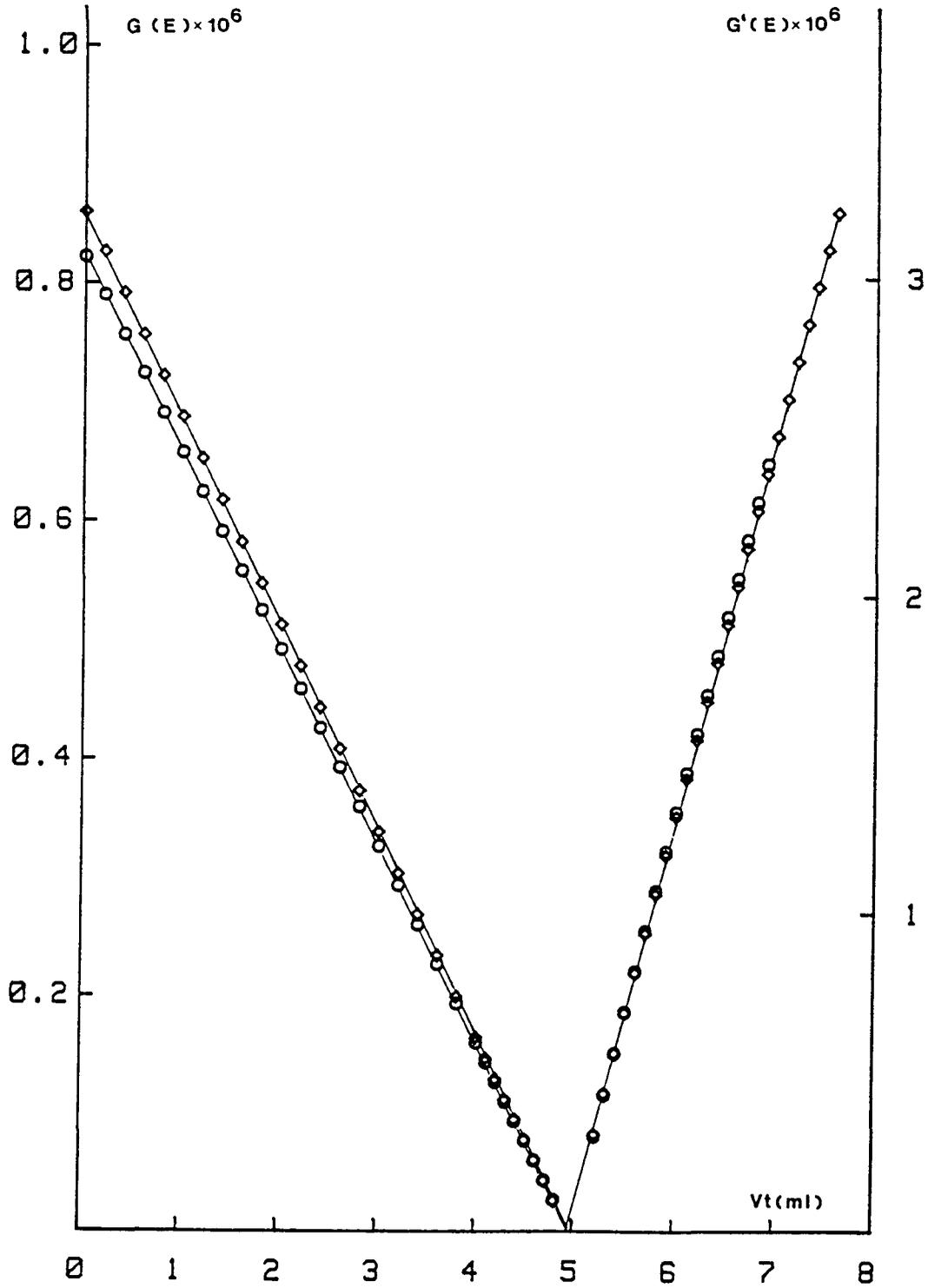


Figura II.1: diagramas de Gran de las valoraciones:
 O: G.1 y Δ : G.2

estas dos valoraciones en las zonas ácida y básica. Se observa una buena concordancia entre los volúmenes equivalentes obtenidos para cada zona y las dos valoraciones, además no se presenta desviación de la linealidad en las rectas, indicando una unión líquida despreciable. La variación en el volumen equivalente medio entre las dos valoraciones no es superior al 0.3 % , siendo en valor absoluto de 0.02 ml y totalmente aceptable ya que es del orden que el fabricante indica como error en la medida para esta bureta con el cilindro de 10 ml.

El valor de pKw, junto con los otros parámetros, está representado en la tabla II.1 para ambas valoraciones:

Tabla II.1: análisis según las funciones de Gran

Val.	<u>zona ácida</u>			<u>zona básica</u>			pKw
	Eó	Veq.	Coef. regr.	E"o	Veq.	Coef. regr.	
G.1	409.30	4.962	1.0000	402.42	4.955	0.9999	13.721
G.2	410.44	4.942	1.0000	401.63	4.926	0.9999	13.727

El tratamiento de estas dos valoraciones con MINIGLASS ha dado prácticamente los mismos resultados. Las figuras II.2 a la II.5 representan las gráficas de curva de valoración y distribución de residuales, separadamente para las zonas ácida y básica. La tabla II.2 muestra los resultados obtenidos mediante el programa tratando independiente y conjuntamente

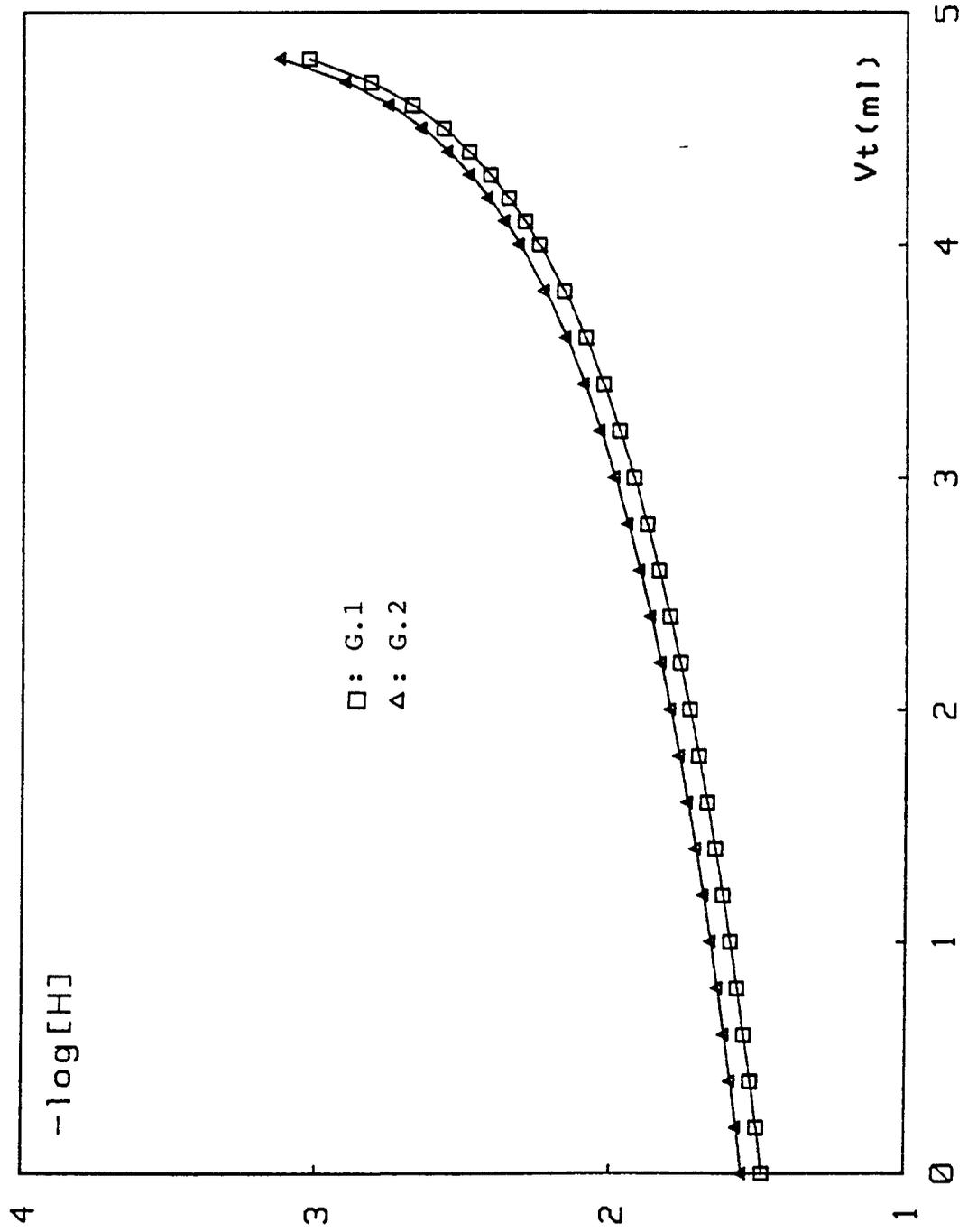


Figura II.2: curvas de valoración del medio iónico (zona ácida)

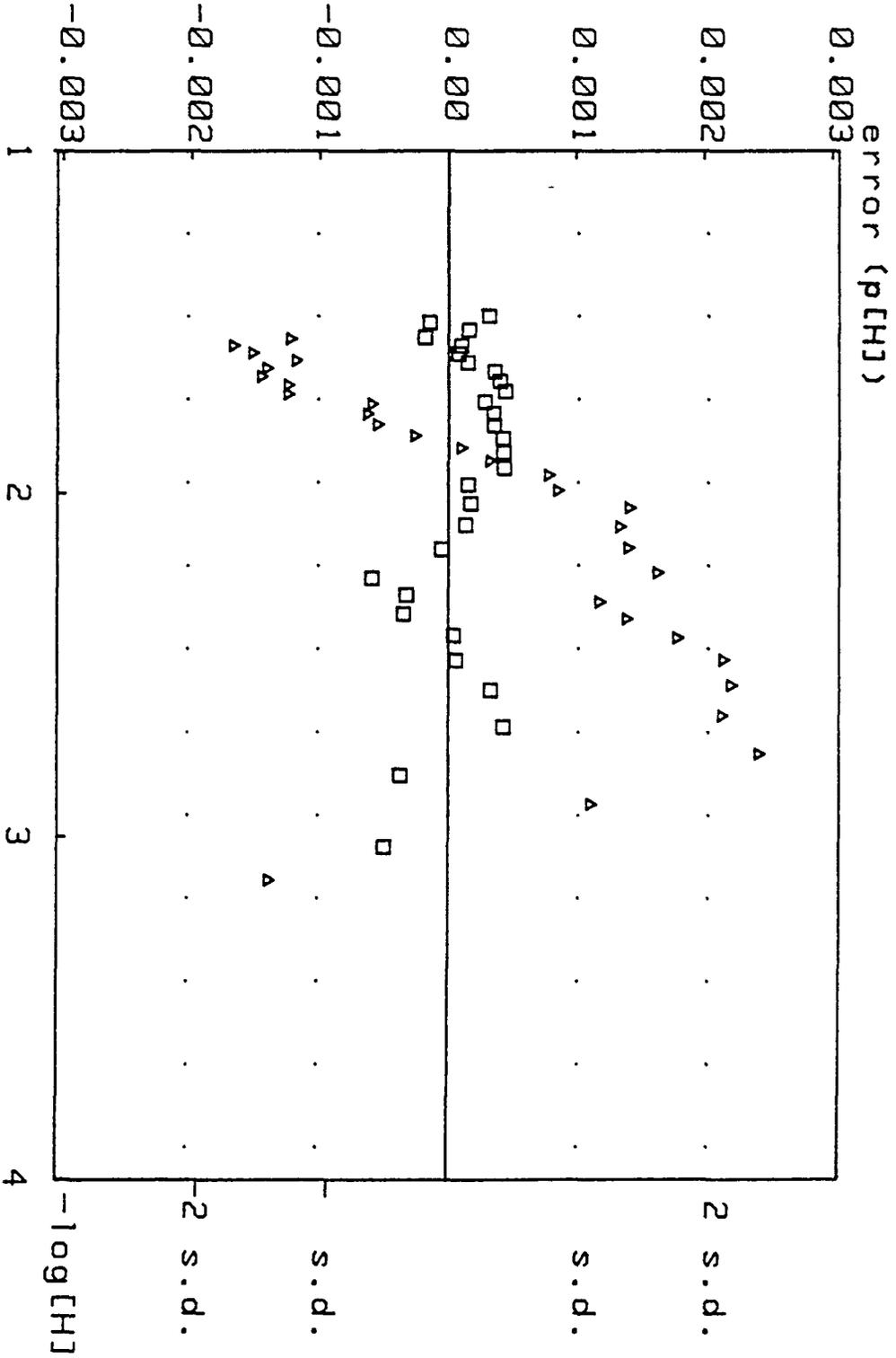


Figura II.3: distribución de los errores en la zona ácida (símbolos como en fig. II.2)

ambas valoraciones.

Tabla II.2: valores calculados por MINIGLASS

(en paréntesis las desviaciones estándar estimadas de los parámetros)

	Eó	pKw	U ^a	s ^b
G.1	409.30(1)	13.721(1)	1.07x10 ⁻⁴	2.66x10 ⁻³
G.2	410.42(2)	13.727(1)	4.56x "	4.45x "
G.1-G.2		13.725(1)	8.03x "	4.48x "

a) suma de los cuadrados de los residuales

b) desviación estándar de los residuales en unidades de pH

El valor de pKw obtenido por nosotros está dentro del intervalo dado en el estudio interlaboratorio (23).

II.7.2.2. Valoraciones P.1 a P.3

Como paso previo a la determinación de las constantes de formación de las especies complejas, se han calculado las constantes de disociación ácido-base de la glicina a partir de los datos experimentales obtenidos en esta serie de valoraciones.

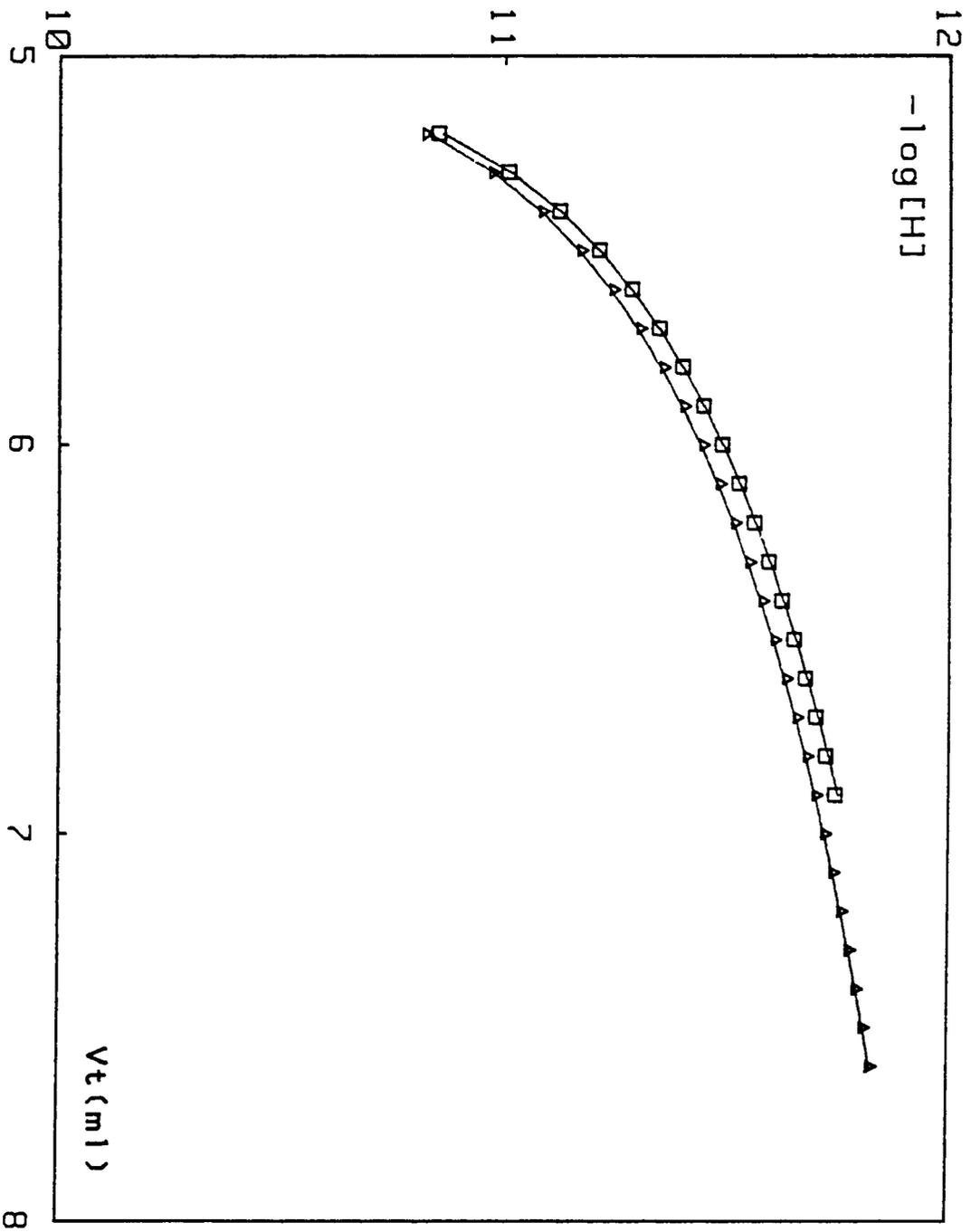


Figura II.4: zona básica de las valoraciones del medio iónico
(símbolos como en fig. II.2)

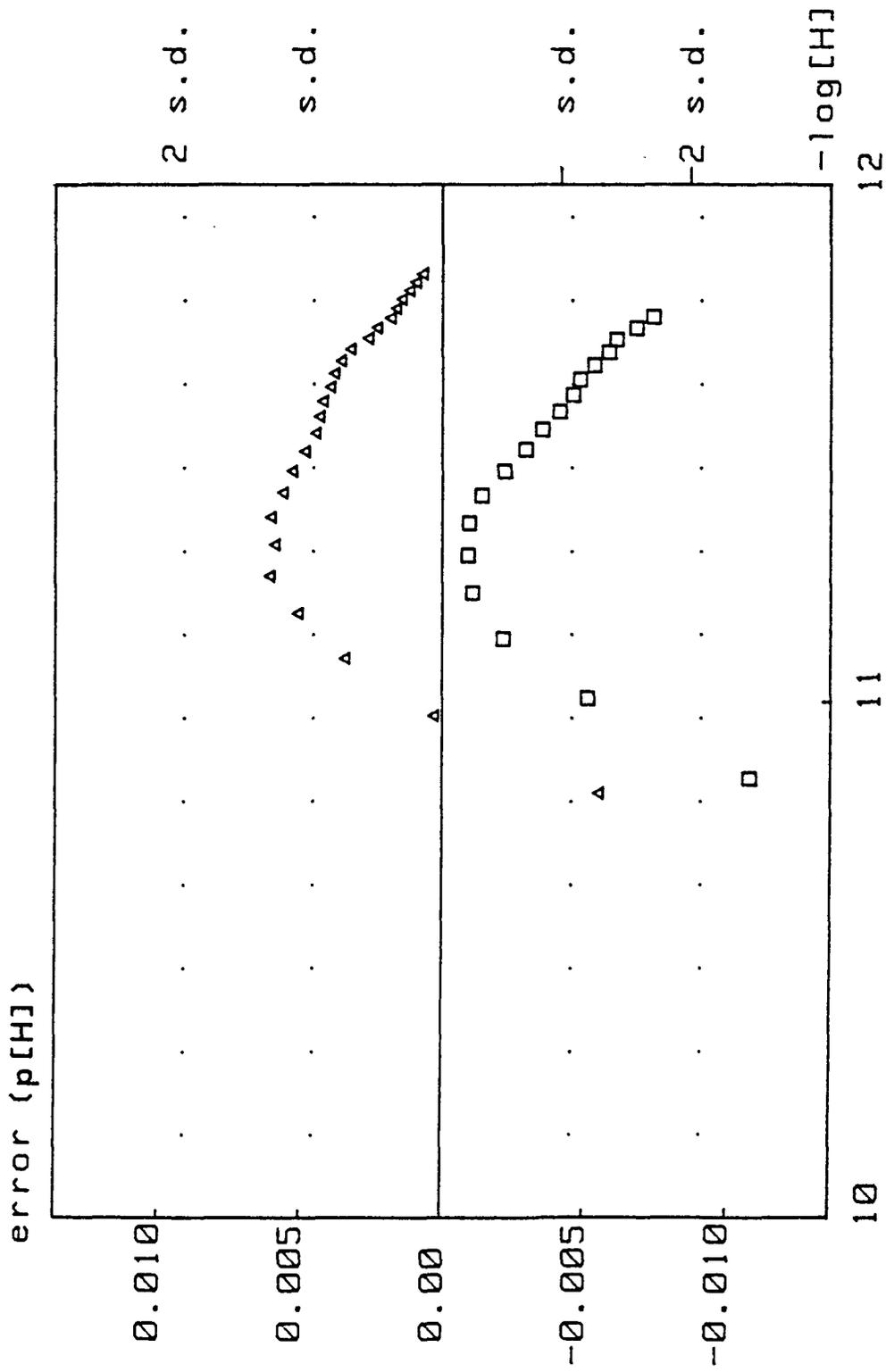


Figura II.5: distribución de los errores en la zona básica (símbolos como en fig. II.2)

El procesamiento de los datos se ha llevado a cabo según los tres procedimientos posibles de cálculo de constantes de ionización, es decir, por el modo PKAS y por el modo COMPLEX (en este caso definido para un sistema con 2 componentes y 2 especies) refinando por Vt y por p(H).

Como aproximación inicial para el cálculo de las constantes se han tomado los valores de p(H) correspondientes a la curva de protonación del anión glicinato en los puntos de $\bar{j}=0.5$ y $\bar{j}=1.5$, que corresponden aproximadamente a unos valores de $\log(K1) = -2.4$ y $\log(K2) = -9.65$ (constantes de disociación), y por tanto: $\log(\beta_{101}) = 9.65$ y $\log(\beta_{102}) = 12.05$. La curva de protonación obtenida experimentalmente se muestra en la figura II.6, junto con la curva teórica según los resultados finales.

Las condiciones experimentales en estas valoraciones están resumidas en la tabla siguiente:

Tabla II.3: condiciones experimentales de P.1-P.3

Valoración	$\left[\text{gly} \right]_{\text{(mM)}}$	intervalo de $-\log [H^+]$
P.1	6.96	1.7 - 10.6
P.2	13.23	1.8 - 10.2
P.3	26.46	2.1 - 10.7

El intervalo de concentraciones escogido corresponde al posteriormente abarcado en la determinación de las

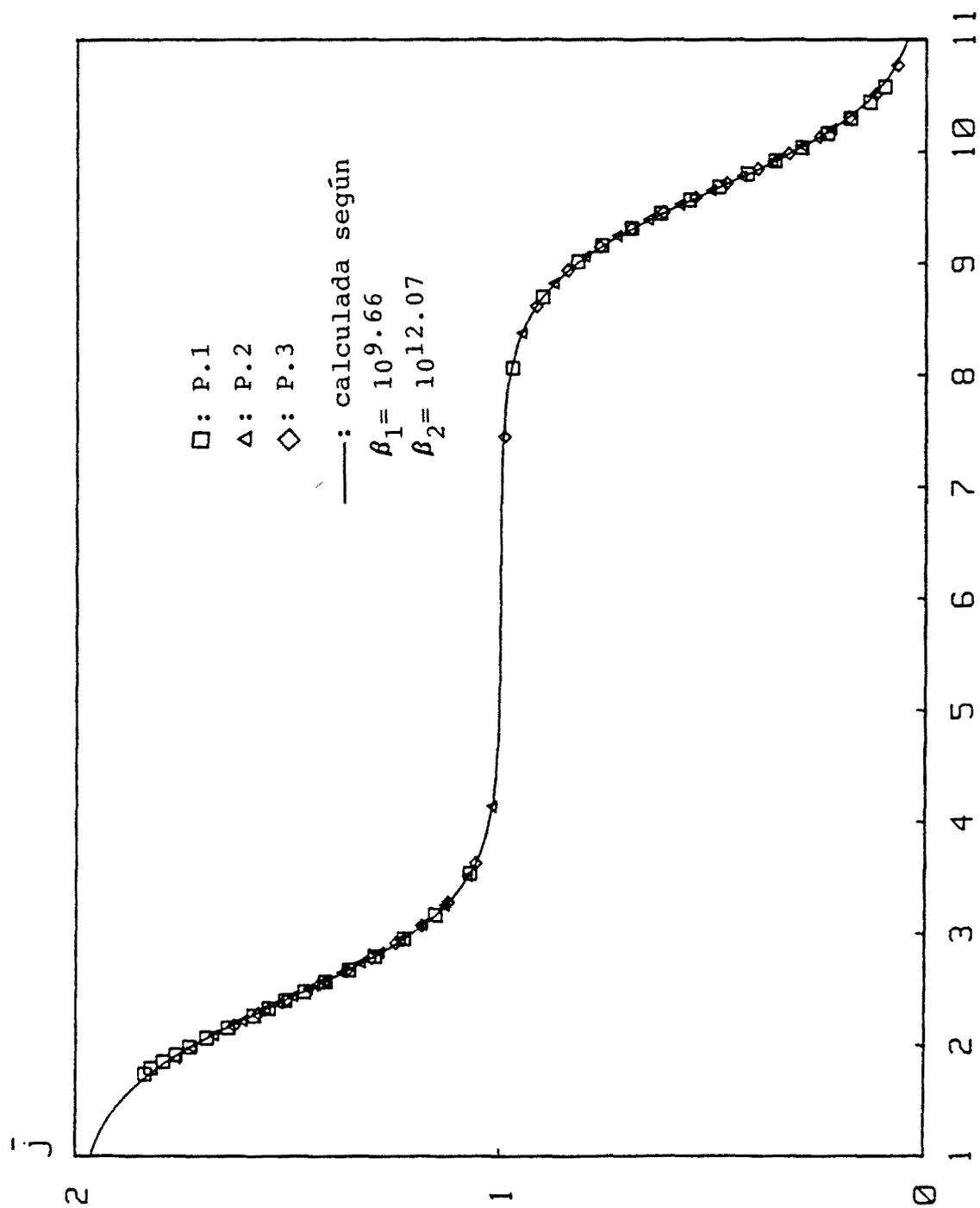


Figura II.6: curva de protonación de la glicina
 (por claridad sólo se representan la mitad de los puntos) $-\log[H^+]$

constantes de complejación. Como se verá a continuación, los valores de las constantes de protonación obtenidos han sido calculados para cada valoración en particular, y después se han tratado conjuntamente, dando valores totalmente equivalentes según los tres métodos de cálculo posibles. En la tabla II.4 se presentan los resultados obtenidos separada y conjuntamente. Los valores que se proponen finalmente se han obtenido promediando todos los valores individuales previos, dando prácticamente los mismos resultados que cuando se han tratado las tres valoraciones conjuntamente.

En las figuras II.7 y II.8 se representan las curvas de valoración y de distribución de los residuales obtenidos por MINIGLASS cuando se ha trabajado en el modo "PKAS".

Con los valores finales obtenidos de las constantes de protonación de la glicina se ha calculado la distribución de sus especies mediante el programa DISTR (apéndice III). La figura II.9 muestra dicha distribución en función del pH.

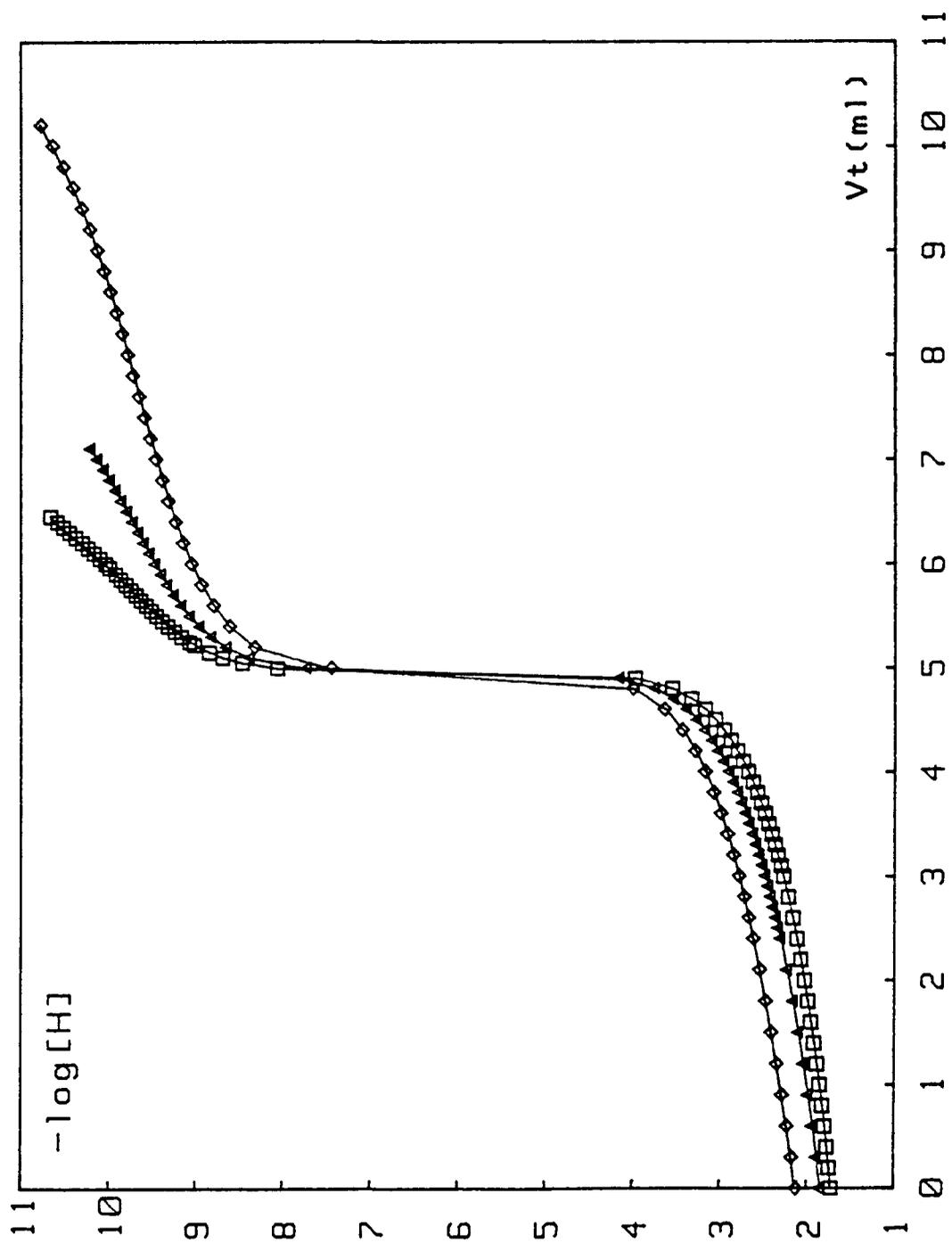


Figura II.7: curvas de valoración experimentales y calculadas de la glicina (símbolos como en fig. II.6)

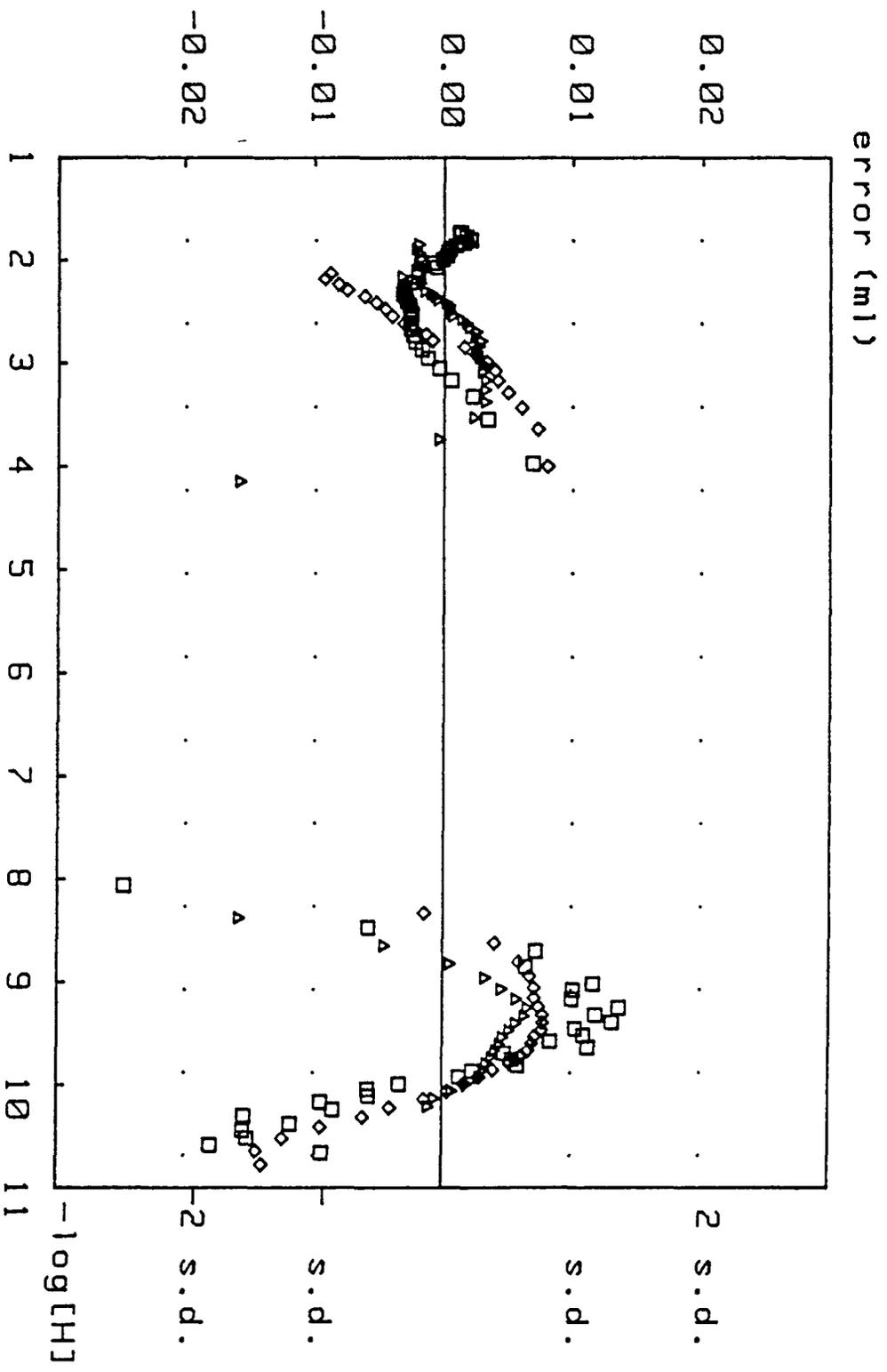


Figura II.8: distribución de errores en las valoraciones de la glicina (símbolos como en fig. II.6)

Tabla II.4: constantes de protonación del ión glicinato

(como $\log \beta_{lmh}$, en paréntesis las desviaciones estándar estimadas en unidades del último dígito)

Valoración	especies:		U	s*
	101	102		
P.1 ^P	9.660(2)	12.076(4)	3.62×10^{-3}	7.64×10^{-3}
P.1 ^v	9.659(2)	12.074(3)	3.81x "	7.84x "
P.1 ^h	9.659(2)	12.075(5)	4.71x "	8.71x "
P.2 ^P	9.652(3)	12.063(5)	7.67x "	1.20×10^{-2}
P.2 ^v	9.657(2)	12.064(2)	3.88x "	8.56×10^{-3}
P.2 ^h	9.652(3)	12.063(5)	7.64x "	1.20×10^{-2}
P.3 ^P	9.657(3)	12.073(5)	5.02x "	1.06x "
P.3 ^v	9.660(2)	12.070(3)	1.02x "	1.51x "
P.3 ^h	9.656(3)	12.072(5)	5.38x "	1.09x "
P.1-P.3 ^P	9.657(4)	12.072(9)	1.71×10^{-2}	1.01x "
P.1-P.3 ^v	9.660(1)	12.069(2)	1.88x "	1.06x "
P.1-P.3 ^h	9.656(2)	12.070(3)	1.83x "	1.05x "

Valores propuestos 9.657(3) 12.070(5)

p: según procedimiento Pkas

v: " " Complex; refine por Vt

h: " " Complex; refine por p(H)

*: la desviación estándar de los residuales se da en unidades de pH para el procedimiento PKAS y el procedimiento Complex en modo pH. En modo Vt se da en ml.

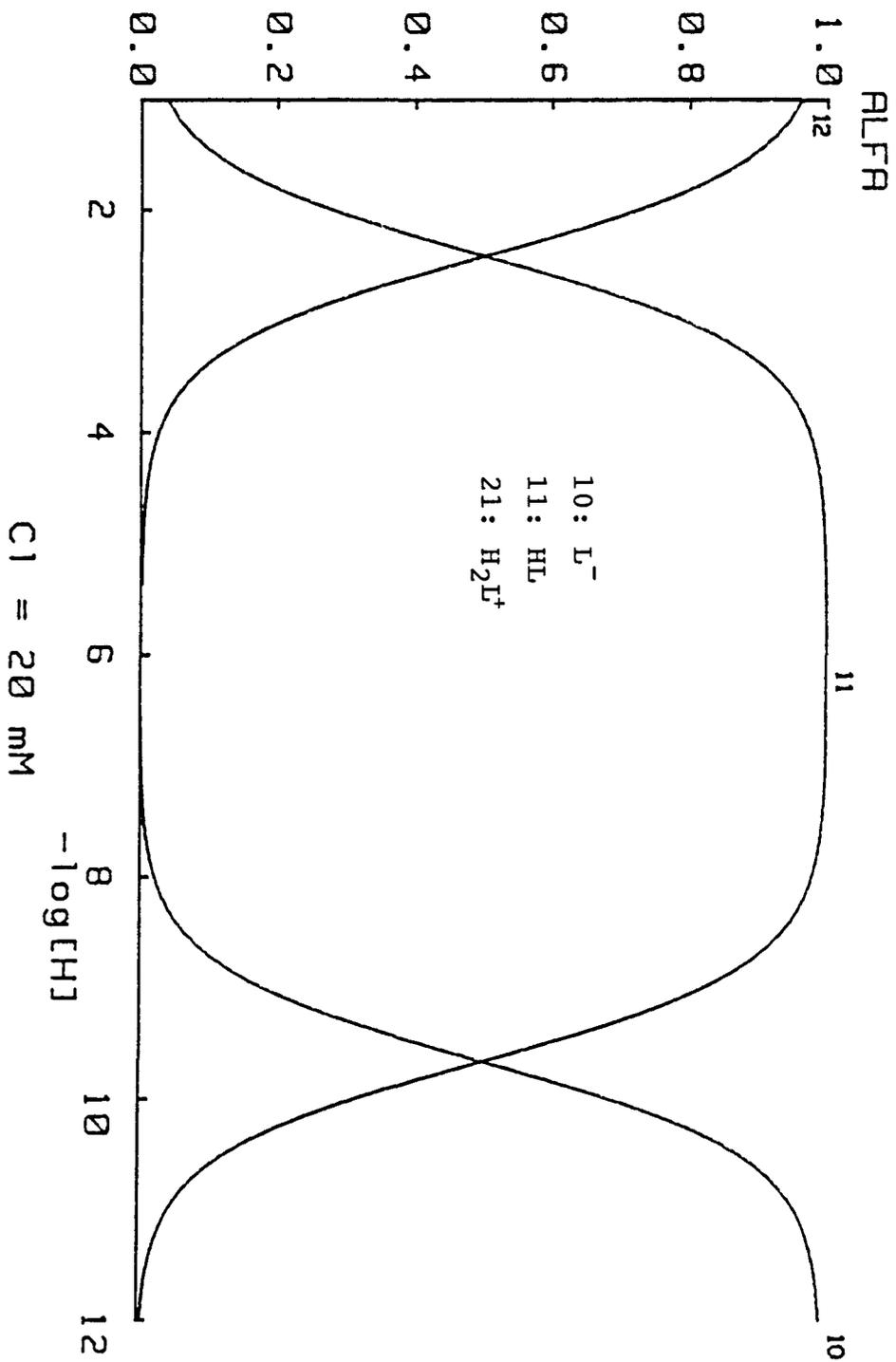


Figura II.9: distribución de especies de la glicina en función del pH

II.7.2.3. Valoraciones C.1 a C.5

Son las correspondientes a la determinación de las constantes de complejación. La estrategia de cálculo seguida ha sido totalmente similar a la del apartado anterior, consistiendo en primer lugar en el trazado de la curva de formación para el sistema Ni(II)-glicina con los datos experimentales (figura II.10) para tener una primera aproximación de las constantes. De los valores de $-\log [\text{gly}^-]$ cuando la variable \bar{n} vale 0.5, 1.5 y 2.5, las sucesivas constantes de formación tienen unos valores aproximados de:

$$\log(K_1)=5.76 \quad \log(K_2)=4.7 \quad \log(K_3)=3.37$$

que corresponden a: $\log(\beta_{110})=5.76$, $\log(\beta_{210})=10.46$, y $\log(\beta_{310})=13.83$.

Análogamente se han tratado los datos experimentales de cada curva independientemente y posteriormente todas a la vez según los modos de refine posibles en el procedimiento Complex.

Antes de pasar al comentario de los resultados, hay que destacar que en el estudio interlaboratorio citado anteriormente, Williams y Corrie (23) postulan, además de los complejos de estequiometría 1:1, 2:1 y 3:1, otras especies hidrolíticas en zonas altas de pH que no obstante, no son apreciadas por otros investigadores en ese mismo estudio. Zuberbuler y Kaden (6) encuentran que la especie Ni(gly)OH debería ser rechazada, mientras que Arnold y col. (24) han comprobado que

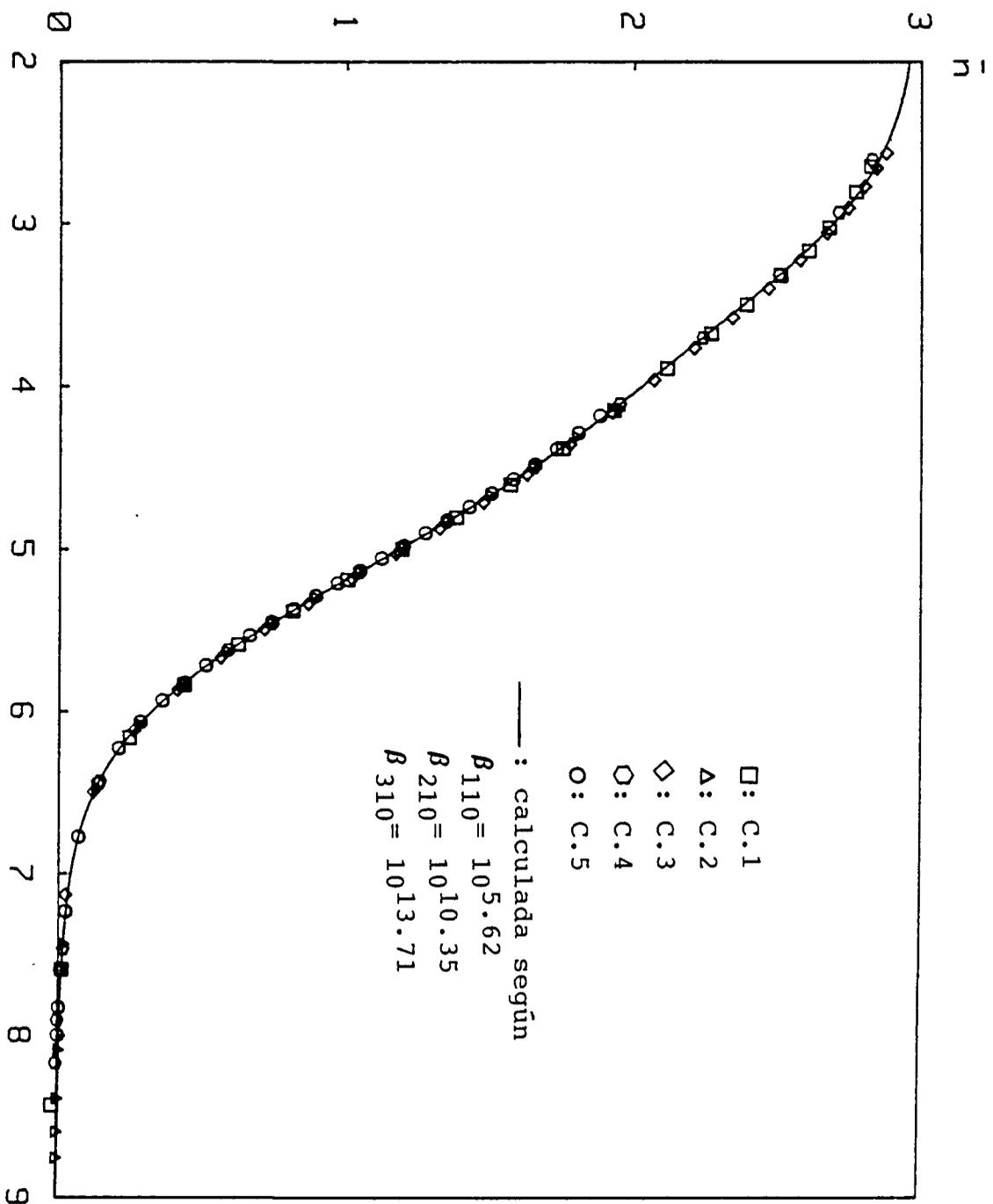


Figura II.10: curva de formación del sistema
Ni(II)-glicina
(representados sólo la mitad de los puntos)

— $\log [A=]$

en algunas valoraciones en particular los resultados se ajustan mejor a los datos experimentales si esa especie es incluida. Por su parte, Gans y col. (18) tampoco la toman en consideración debido a la zona de pH de trabajo.

En nuestro caso, tampoco se ha tenido en cuenta esta especie hidrolítica, ya que como máximo se ha trabajado a un p(H) de 9.7 , donde es minoritaria, y además su inclusión en el modelo no mejoraba de manera apreciable los resultados obtenidos.

Las condiciones experimentales que corresponden a este grupo de valoraciones se resumen en la tabla II.5. Como comentario digamos que el límite superior de p(H) es debido a que por encima de los valores indicados, las lecturas de potencial empezaban a oscilar y hacerse inestables, probablemente debido a una incipiente precipitación ó bien a la formación de especies hidrolíticas que no se han considerado en los cálculos. Como se puede ver en la tabla, este límite inferior es más pequeño cuanto mayor es la concentración de Ni(II) y menor la relación ligando/metal.

Tabla II.5: condiciones experimentales en las valoraciones C.1 a C.5

Val.	$[\text{gly}]_0$ (mM)	$[\text{Ni}^{2+}]_0$ (mM)	relación ligando/metal	intervalo de $-\log[\text{H}^+]$
C.1	12.02	2.49	4.83	2.9-9.6
C.2	10.80	5.33	2.02	2.9-9
C.3	21.60	5.33	4.05	3.6-9.7
C.4	33.29	8.71	3.82	2.9-9.6
C.5	24.05	11.88	2.02	3.3-8.3

Un primer análisis visual de la curva de formación de este sistema nos indica que las cinco diferentes valoraciones pasan por la misma curva, no apreciándose desviaciones hasta un valor de \bar{n} cercano a 3, debido a las razones previamente comentadas. De todas maneras, podemos concluir que en nuestro caso se pueden despreciar en principio especies mixtas que incluyan H^+ ó OH^- .

El tratamiento numérico de este conjunto de valoraciones se representa en la siguiente tabla:

Tabla II.6: Constantes de complejación obtenidas
para el sistema Ni(II)-glicina

(como $\log \beta_{lmh}$)

Valoración	especies			U	s
	110	210	310		
C.1 ^v	5.623(6)	10.356(5)	13.729(7)	3.59×10^{-4}	3.16×10^{-3}
C.1 ^h	5.624(3)	10.356(4)	13.731(5)	1.15×10^{-3}	5.64x "
C.2 ^v	5.619(3)	10.338(3)	13.710(21)	3.84×10^{-4}	3.46x "
C.2 ^h	5.625(2)	10.336(3)	13.740(11)	9.26x "	5.38x "
C.3 ^v	5.625(3)	10.367(3)	13.750(3)	4.47x "	3.08x "
C.3 ^h	5.619(3)	10.364(4)	13.745(5)	2.01×10^{-3}	6.53x "
C.4 ^v	5.617(7)	10.340(6)	13.689(8)	1.11x "	6.65x "
C.4 ^h	5.621(5)	10.343(6)	13.693(7)	1.63x "	8.08x "
C.5 ^v	5.611(1)	10.328(1)	13.646(8)	4.98×10^{-4}	3.04x "
C.5 ^h	5.613(1)	10.329(2)	13.646(7)	5.51x"	3.19x "
C.1-C.5 ^v	5.615(2)	10.333(2)	13.709(3)	1.34×10^{-2}	7.99x "
C.1-C.5 ^h	5.619(2)	10.341(2)	13.715(3)	1.87x "	9.42x "
Valores propuestos	5.620(5)	10.346(14)	13.708(39)		

v: refine según Vt
h: " según p(H)

(desviaciones estándar de los residuales como en tabla II.4)

Como podemos observar de estos resultados, los valores obtenidos por ambos métodos de refine resultan ser virtualmente los mismos. Los resultados de cada valoración calculados independiente según V_t ó $p(H)$ tienen diferencias del orden de milésimas exceptuando el caso de la tercera constante en la valoración C.2, que además es la que presenta unas mayores desviaciones estándar. En el tratamiento conjunto de todas las valoraciones vemos que la segunda constante de formación es la que presenta mayor diferencia, aunque ésta es menor que una centésima en la magnitud de $\log(\beta_{210})$ y corresponde a una diferencia de un 1.8 % en el valor de la constante propiamente dicha (β_{210}).

Los valores propuestos al final de la tabla II.6 están calculados promediando todos los resultados individuales para cada constante. Estos valores están en conjunto más cerca de los resultados obtenidos al refinar las cinco valoraciones respecto a $p(H)$, aunque de todas maneras, las diferencias entre las constantes obtenidas por V_t , $p(H)$ ó haciendo las medias de los valores independientes son pequeñas y no consideramos que sean significativas. Como vemos en la figura II.10, se ha conseguido una buena concordancia entre las curvas de formación experimentales y la teórica calculada a partir de las constantes propuestas en la tabla II.6.

En las figuras II.11 y II.12 se representan las gráficas generadas por el programa de las curvas de valoración y de distribución de los residuales cuando el refinamiento fue efectuado de las cinco valoraciones simultáneamente, y respecto al volumen de valorante. Si nos fijamos en la distribución de los residuales, vemos que la valoración que presenta una mayor desviación es la curva C.3, que en la tabla II.6 es la que presenta un valor de $\log(\beta_{210})$ que más se aparta del resto.

La distribución de especies en función del pH para el sistema Ni(II)-glicina se representa en la figura II.13, en base resultados de sus constantes de formación.

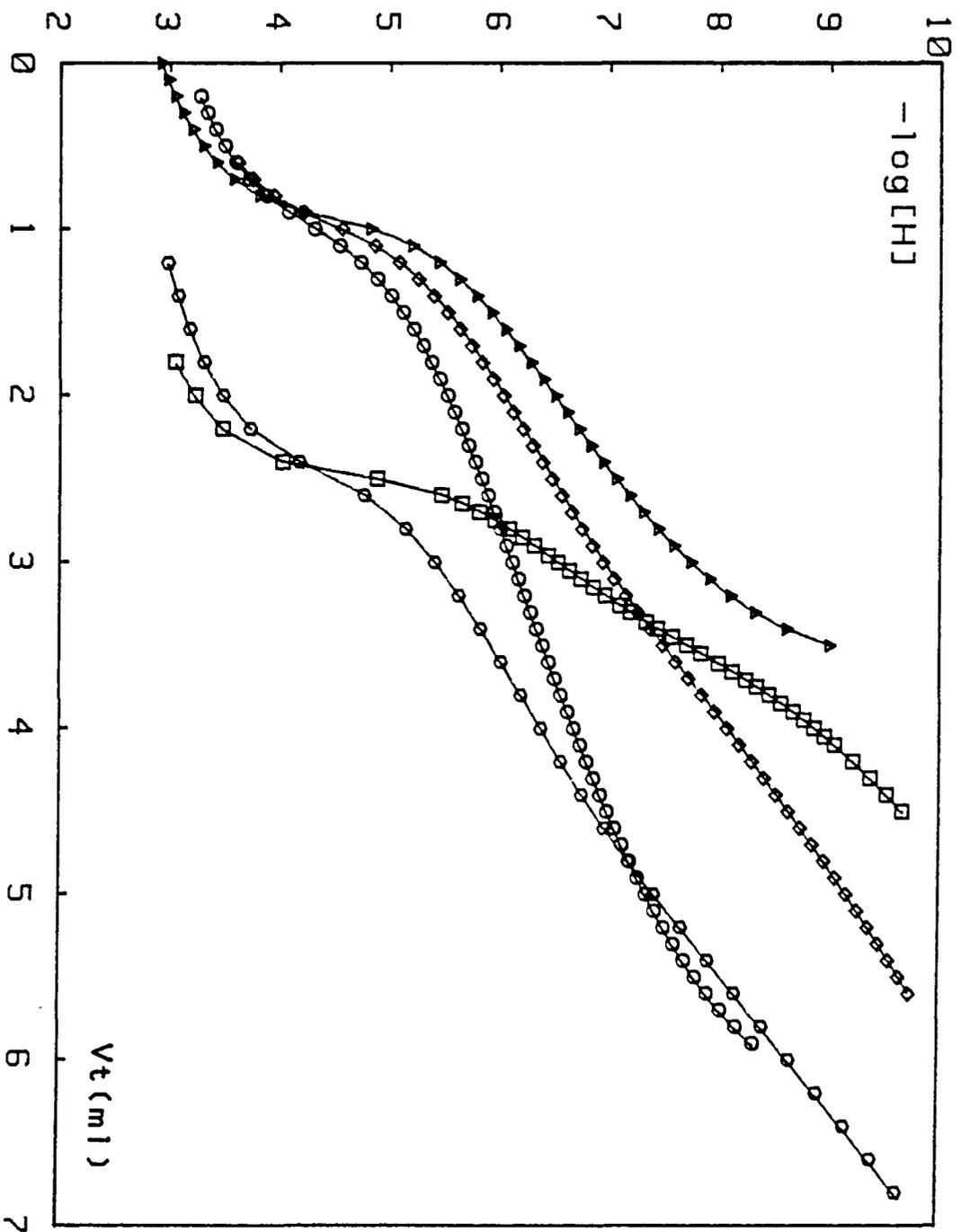


Figura II.11: curvas de valoración experimentales y calculadas del sistema Ni(II)-glicina (símbolos como fig. II.10)

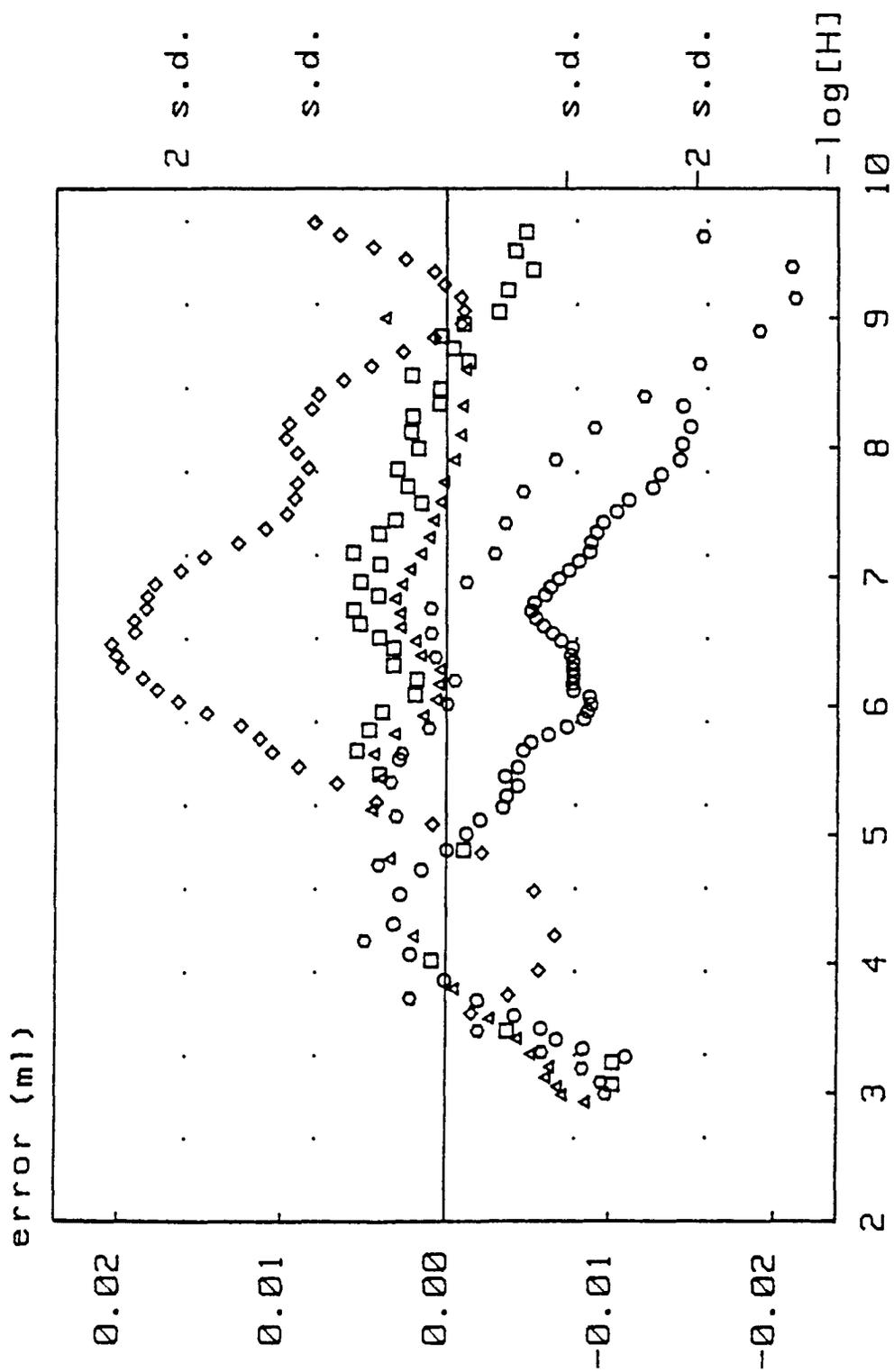


Figura II.12: distribución de errores en las valoraciones C.1 a C.5
(símbolos como en fig. II.10)

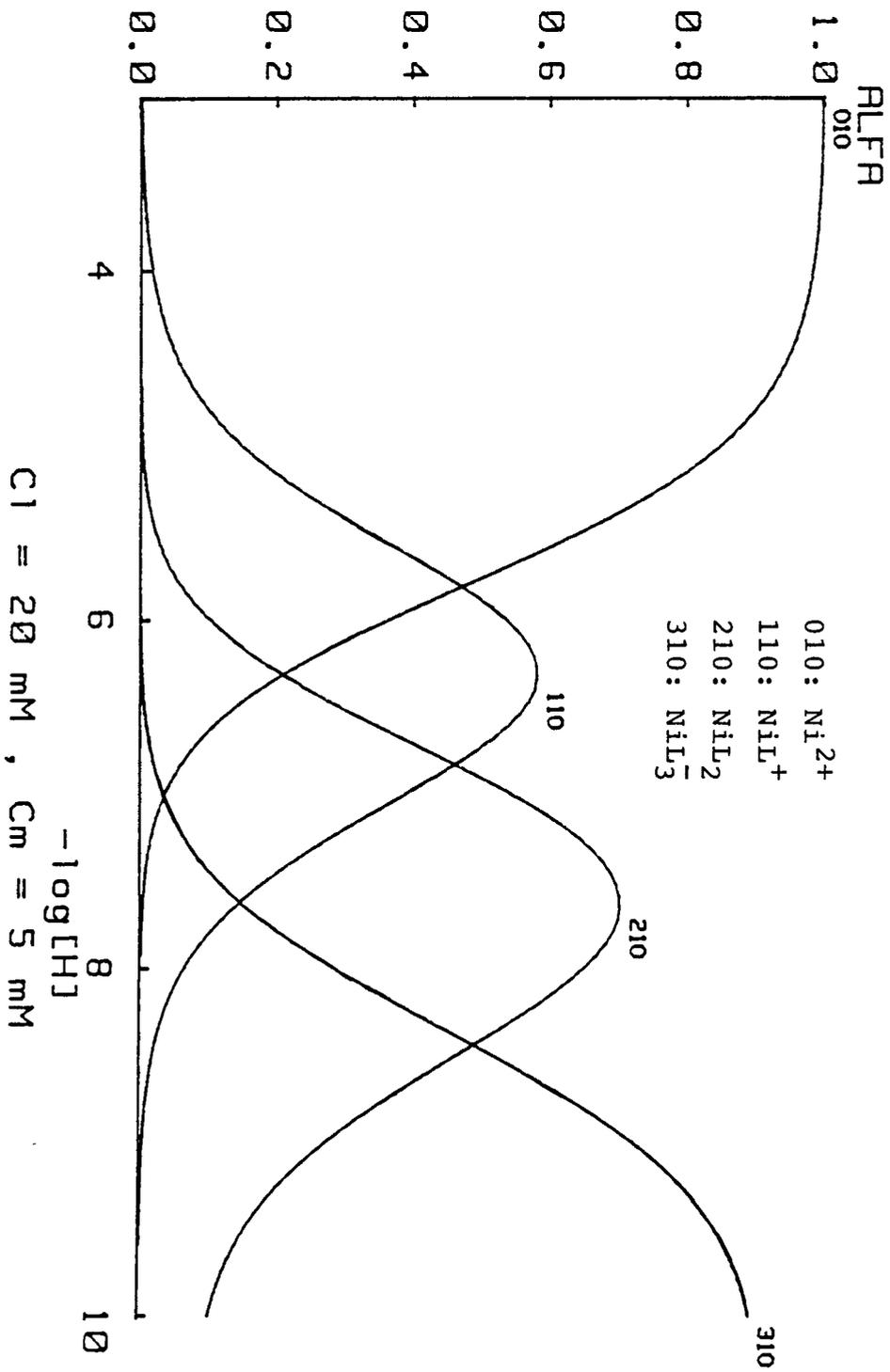


Figura II.13: distribución de especies de Ni²⁺ en el sistema Ni(II)-glicina

II.7.3. Comentarios al cálculo de las constantes

Además de presentarse los resultados en el apartado anterior, se ha considerado conveniente "ilustrar" gráficamente cómo funciona en la práctica el método de refinamiento de las constantes calculadas y la caracterización de las diferentes especies en equilibrio.

Para que estos apartados se vean reflejados de una manera sencilla se ha trabajado sobre una sólo curva de valoración en cada caso, correspondientes a la determinación de los pK_a de la glicina, y a sus complejos con Ni(II), respectivamente.

En el primer caso se han tomado los datos experimentales correspondientes a la valoración P.1, a partir de la cual se han refinado las constantes de protonación de la glicina. Los valores iniciales de éstas constantes se han supuesto dando unos valores de $pK_a = 5$ para el grupo carboxílico, y de 9 para el grupo amino, que nos dan unos valores de $\log(\beta_{110}) = 9$, y de $\log(\beta_{210}) = 14$.

En realidad estos valores son muy diferentes de los reales, ya que la diferencia respecto a la primera constante de protonación es mayor de 0.6 unidades (logarítmicas), y respecto a la segunda es casi de 2 unidades.

Estos valores iniciales se han refinado utilizando el procedimiento Pkas de MINIGLASS en cuatro iteraciones, cuyos

resultados parciales han sido:

	s ($p(H)$)	$\log(\beta_{110})$	$\log(\beta_{210})$
datos iniciales	1.06	9.000	14.000
primera iteración	0.119	9.655	12.492
segunda "	0.0138	9.660	12.122
tercera "	0.00362	9.660	12.076
cuarta "	0.00362	9.660	12.076

La eficiencia del método se puede deducir de estos resultados, ya que la desviación estándar de los residuales (s) decrece aproximadamente en un orden de magnitud en cada ciclo, hasta llegar al valor definitivo ya en la tercera iteración.

Este proceso se puede visualizar en la figura II.14, en la que se representan las curvas de valoración calculadas (trazo continuo) y las experimentales (cuadrados), para los datos de partida y los resultados intermedios de los valores de las constantes. En la figura se aprecia claramente cómo se va ajustando progresivamente la curva calculada a la experimental, siendo más rápido el ajuste en la zona básica que en la ácida, ya que el valor de la primera constante de protonación supuesto era más cercano al real que en la segunda. En la figura no se representan las curvas de la cuarta iteración, ya que ha dado un resultado idéntico a la tercera.

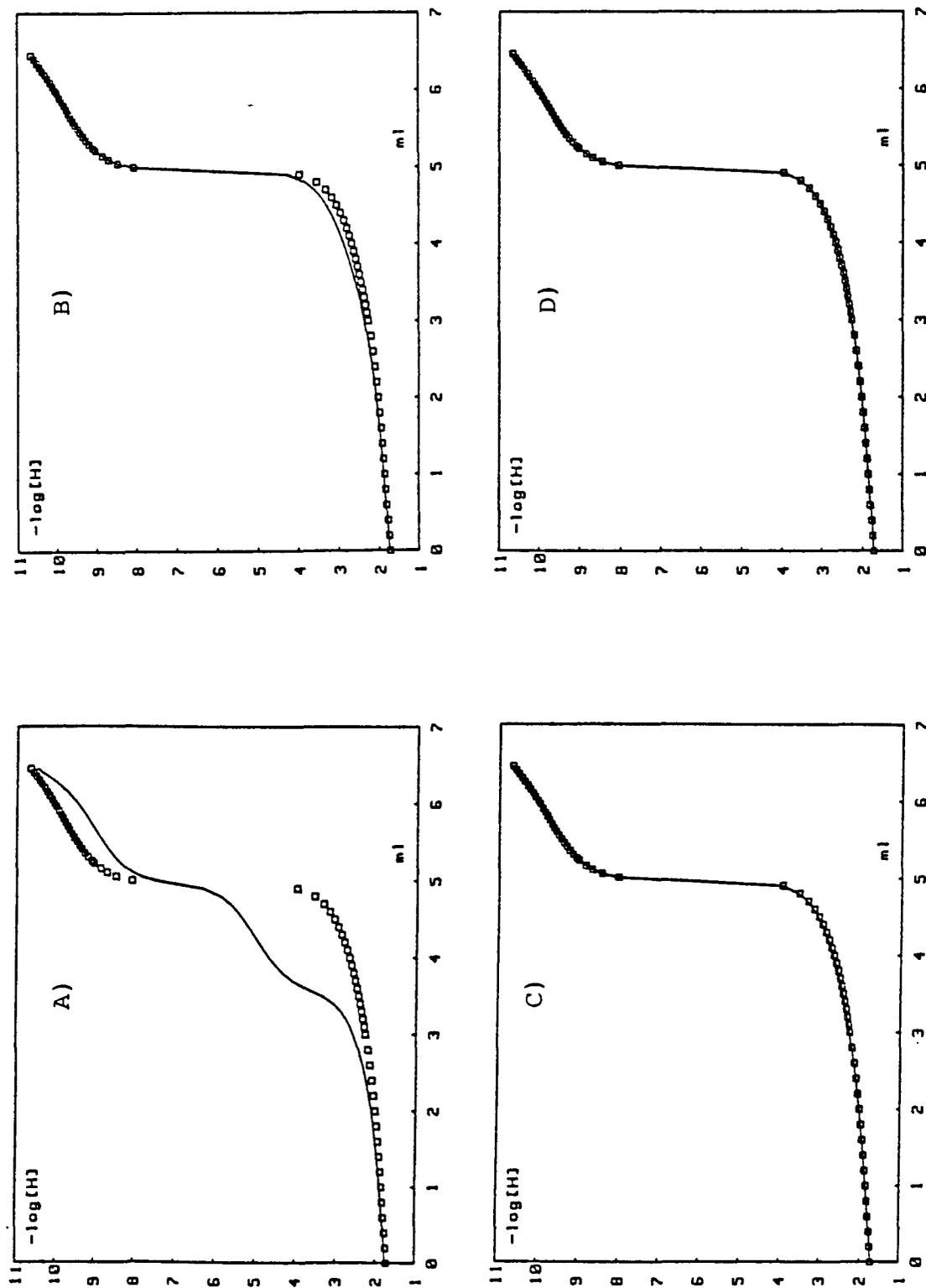


Figura II.14: proceso de refine de las constantes de protonación de la glicina
A: datos iniciales; B, C y D: primer, segundo y tercer ciclo

El segundo caso en estudio se ha llevado a cabo con los datos correspondientes a la valoración C.3, para comprobar el efecto de la adición de sucesivas especies al modelo de complejación del sistema Ni(II)-glicina. En primer lugar se ha calculado la curva de valoración suponiendo que no existía complejación, y posteriormente se han añadido sucesivamente al modelo las especies 110, 210 y 310, refinando las constantes de complejación en cada paso. En este caso se ha trabajado en el modo Complex de MINIGLASS, refinando las constantes por el volumen de valorante añadido. Los pasos sucesivos se resumen a continuación:

datos de entrada ^a	N ^b	valores refinados ^c	s (ml)
1) -	-	-	1.90
2) 110 (4)	3	6.018(553)	1.08
3) 110 (6.018)	5	5.613(233)	0.391
210 (12)		10.508(174)	
4) 110 (5.613)	7	5.625(3)	0.0031
210 (10.508)		10.367(3)	
310 (15.4)		13.750(3)	

a: como especies l:m:h , en paréntesis $\log(\beta_{lmh})$.

b: número de iteraciones al refinar las constantes.

c: valores de $\log(\beta_{lmh})$, en paréntesis su desviación estándar en unidades del último dígito.

El paso 1) corresponde al supuesto de que no existe complejación. En el paso 2) se considera la formación de una única especie, de estequiometría 110, a la que se ha asignado arbitrariamente una constante de formación de 10^4 , que una vez refinada ha dado un valor de $\log(\beta_{110}) = 6.02$. Este valor se ha introducido como dato de entrada para el paso 3). En éste se ha supuesto además la formación de la especie 210, cuya constante de formación se ha supuesto como :

$\log(\beta_{210}) = 2 \times \log(\beta_{110})$. Para el paso 4) se han introducido los valores refinados de las primeras constantes, mientras que el valor estimado de β_{310} se ha calculado según:

$$\log(\beta_{310}) = \log(\beta_{210}) + \log K_2.$$

En la figura II.15 se aprecian los pasos sucesivos, representándose las curvas de valoración calculadas y experimentales a partir de las constantes refinadas de cada paso. En ella observamos la gran diferencia que existe cuando no se supone complejación, y cómo se va ajustando la curva teórica a la experimental al introducir las sucesivas especies al modelo.

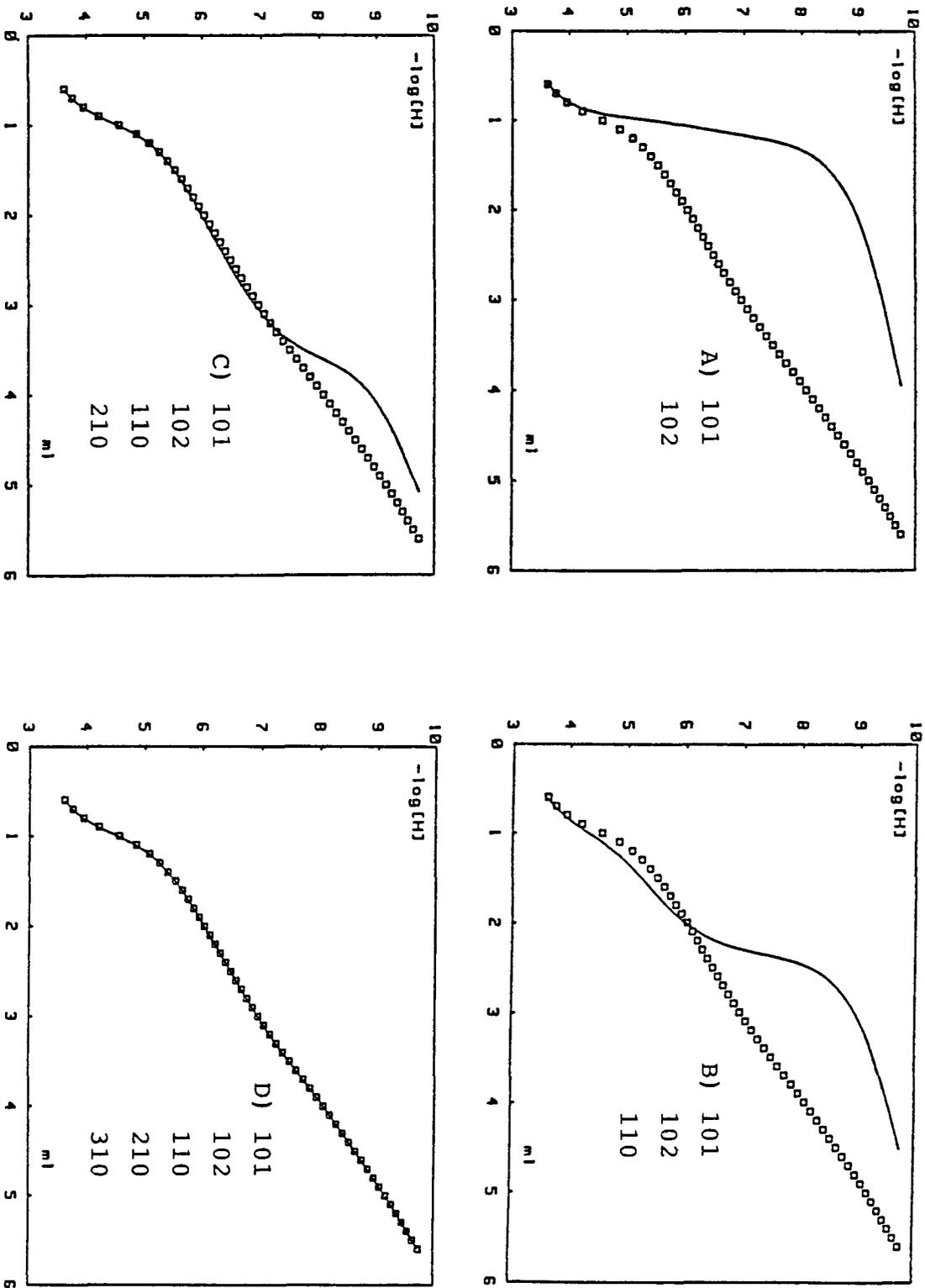


Figura II.15: curvas de valoración calculadas y experimentales del sistema Ni(II)-glicina para diferentes especies propuestas (como l,m,h)

II.8. Discusión

En primer lugar hay que decir que tanto el sistema de adquisición de datos y el programa construido para su posterior tratamiento han funcionado a nuestra entera satisfacción, además de ser ambos totalmente fiables.

La aplicación a un sistema ampliamente estudiado, como lo es el de Ni(II)-glicina- H^+ nos ha permitido comparar los resultados obtenidos por nosotros con los datos de la bibliografía reciente. La comparación se representa en la tabla II.7, donde vemos que nuestros resultados presentan buena concordancia con los obtenidos por otros investigadores. Hay que recordar, además, que aparte de los datos correspondientes al estudio interlaboratorio (23, 25), las otras citas se refieren a programas de cálculo de constantes de estabilidad (6, 18) y a un sistema de valoración automática (24), por lo que nuestra elección al escoger este sistema en particular se ve plenamente justificada.

De esta comparación se ha comprobado que los resultados obtenidos están en la línea de los conseguidos por el resto de investigadores citados, confirmando el buen funcionamiento del sistema automático de valoración y el programa MINIGLASS.

Adicionalmente, hay que resaltar el hecho de que las desviaciones estándar obtenidas al promediar los resultados de cada valoración independientemente son prácticamente las mismas que las que se dan en las referencias (18) y (24). Esto indica que los errores en los distintos resultados también son comparables, y confirma que el estudio del sistema Ni(II)-glicina como "test" para sistemas de valoración y tratamiento de resultados se puede considerar como un estándar (para una descripción más amplia de la distribución de los errores en el estudio interlaboratorio ver la referencia 25).

Tabla II.7: comparación de resultados

(sistema Ni(II)-glicina- H^+ , a $25^{\circ}C$, $I=1$ M en NaCl)

especies (l, m, h)	referencias:				
	6	23,25	18	24	esta memoria
	$\log(\beta_{lmh})$				
101	9.63	9.651(12)	9.645(3)	9.621	9.657(3)
102	12.08	12.071(26)	12.060(6)	12.064	12.070(5)
110	5.58	5.615(35)	5.629(5)	5.550(7)	5.620(5)
210	10.30	10.363(62)	10.397(13)	10.251(12)	10.346(14)
310	13.75	13.930(340)	13.846(41)	13.750(30)	13.708(39)
pKw	13.69	13.693(20)	--	13.693	13.715(12)

II.9. Listado del programa MINIGLASS

```

$ref 70$
PROGRAM miniglass(input,output,listing);
import dgl_lib;

CONST mil=1000;
pi=3.1416;
mxt=9 (max. número de valoraciones);
mxp=900 (número max. de puntos);
ppt=100(puntos en una valoración);
tcomp=4 (número máximo de componentes);

TYPE
sixpsix=array[1..mxt,1..mxt] of real;
granmat=array[1..mxt,1..ppt] of real;
ptr=^granmat;
matsec=array[1..mxt,1..mxp] of real;
qtr=^matsec;
special=array[1..mxt,1..ppt,1..tcomp] of real;
catch=^special;
fortmat=array[1..mxp] of real;
stre=^fortmat;
archiv=string[20];

VAR
listing:text;
f,model:text;
fmodel:archiv;
fname:array[1..mxt] of archiv;
ncomp,i,j,ntit,nesp:INTEGER;
flag1,flag2,flag5:char;
volume,emf,dif,ph,phcor,secdif,volcor,h: ptr;
tot,cl:catch;
pl:array[1..tcomp] of integer;
coef:ARRAY[1..20,1..tcomp] of integer;
deriv: qtr;
error: stre;
param:ARRAY[1..90] of real;
beta:array[1..20] of real;
total,npref,totalpoints:integer;
shift,qs,sh,vo:array[1..mxt] of real;
first,last,npoints:array[1..mxt] of integer;
eo,sl,ht,tho,sign,hsig:array[1..mxt] of real;
co:array[1..mxt,1..tcomp] of real;
factor,ix,iy,sx,sy,qres,qres0,qres1,qres2,qres3,qres4,
qres5,stdev,pkw:real;

```

```
kw,jh,joh,phmin,phmax,volmax,volmin:real;
acabado,selector,switchcon,com:boolean;
```

```
function alog(l1:real):real;
var l11:real;
begin l11:=exp(2.302585*l1);
alog:=l11;
end;
```

```
function lgt(l2:real):real;
var l12:real;
begin l12:=0.434294*ln(l2);
lgt:=l12;
end;
```

```
function RS(R:real;K:integer):real;
var L:integer;
U:real;
begin if K<0 then U:=1/R;
if K=0 then U:=1 else if K=1 then U:=R
else begin L:=abs(K);
U:=1;
repeat if L mod 2 =1 then U:=R*U;
L:=L div 2;
if L<>0 then R:=R*R
until L=0;
if K<0 then U:=1/U
end;
RS:=U;
end;
```

```
function signo(sg:real):integer;
VAR sn:integer;
begin if sg>0 then
sn:=1 else if sg<0 then sn:=-1 else sn:=0;
signo:=sn;
end;
```

```
function residuals:real;
var rt:integer;
ttrs:real;
begin ttrs:=0;
for rt:=1 to ntit do begin
ttrs:=ttrs+qs[rt];
end;
```

```

residuals:=ttrs;
end;

function stv:real;
var stan:real;
begin stan:=sqrt(residuals/(total-1-npref));
stv:=stan;
end;

function prime(tn,pn:integer;H:real):real;
VAR ab,ks,ns:integer;
pbeta:real;
begin pbeta:=0;
for ks:=1 to nesp do begin ab:=0;
for ns:=1 to ncomp do begin
if (ns<>pn) and (ns<>ncomp) then
ab:=ab+abs(coef[ks,ns]);
end;
if ab=0 then begin
if (coef[ks,pn]< >0) and (coef[ks,ncomp]<>0) then
pbeta:=pbeta +alog(beta[ks])*RS(H,coef[ks,ncomp]);
end;
end;
prime:=co[tn,pn]/(pbeta+1);
end;

procedure matinv(rne:sixpsix;f1:integer;var enr:sixpsix);
var ik,k1,l2:integer;
va:real;
begin for ik:=1 to f1 do begin
for k1:=1 to f1 do begin
enr[ik,k1]:=0;
enr[k1,ik]:=0;
end;
enr[ik,ik]:=1.0;
end;
enr[f1,f1]:=1.0;
for k1:=1 to f1 do begin
va:= rne[k1,k1];
if abs(va)<=1.0E-50 then va:=1.0E-10;
for ik:=1 to f1 do begin
rne[k1,ik]:=rne[k1,ik]/va;
enr[k1,ik]:=enr[k1,ik]/va;
end;
for l2:=1 to f1 do begin

```

```

if l2<>k1 then begin
va:=rne[l2,k1];
for ik:=1 to f1 do begin
rne[l2,ik]:=rne[l2,ik]-va*rne[k1,ik];
enr[l2,ik]:=enr[l2,ik]-va*enr[k1,ik];
end;
end;
end;
end;
end;

procedure gran(j4:integer);
var i4:integer;
prot,c1:real;
begin
qs[j4]:=0;
for i4:=first[j4] to last[j4] do begin
c1:=tot^[j4,i4,1];
prot:=0.5*(c1+sqrt(c1*c1+4*kw));
phcor^[j4,i4]:=-lgt(prot);
dif^[j4,i4]:=ph^[j4,i4]-phcor^[j4,i4];
qs[j4]:=qs[j4]+sqr(dif^[j4,i4]);
end;
end;

procedure setconc(kt:integer);
var kp,k1,ic:integer;
vsol,prs:real;
begin
prs:=0;
if flag1<>'G' then begin
for k1:=1 to (ncomp-1) do begin
for kp:=1 to npoints[kt] do
tot^[kt,kp,k1]:=co[kt,k1]*vo[kt]/(vo[kt]+volume^[kt,kp]);
end;
end;
if ncomp>1 then begin
for ic:=1 to ncomp-1 do prs:=prs+pl[ic]*co[kt,ic];
end;
tho[kt]:=prs+hsig[kt]*co[kt,ncomp];
for kp:=1 to npoints[kt] do begin
vsol:=volume^[kt,kp];
tot^[kt,kp,ncomp]:=(tho[kt]*vo[kt]+ht[kt]*vsol)/(vo[kt]+vsol);
end;
end;
end;

```

```

procedure pkas(i3:integer);
const epsilon=1.0e-4;
var b,b1,pr,dr,bt,bv,dbt,dbv,vto,vr,toth,totl:real;
j3,k3,l3:integer;
begin
qs[i3]:=0;
for j3:=first[i3] to last[i3] do begin
b:=h^[i3,j3];
toth:=tot^[i3,j3,2];
totl:=tot^[i3,j3,1];
repeat
bt:=0;
bv:=0;
dbt:=0;
dbv:=0;
for k3:=1 to nesp do begin
vr:=RS(b,k3)*alog(beta[k3]);
bt:=bt+k3*vr;
bv:=bv+vr;
dbt:=dbt+k3*k3*vr;
end;
bv:=bv+1;
pr:=bt/bv-(toth-b+kw/b)/totl;
dr:=(b+kw/b)/totl+(dbt*bv-bt*bt)/(bv*bv);
b1:=b;
b:=b1/exp(pr/dr);
until abs(pr)<epsilon;
phcor^[i3,j3]:=-lgt(b);
dif^[i3,j3]:=ph^[i3,j3]-phcor^[i3,j3];
qs[i3]:=qs[i3]+sqr(dif^[i3,j3]);
end;
end;

procedure outbeta(ob:integer);
var ta:integer;
begin param[ob]:=0;
if ob=nesp then nesp:=nesp-1 else begin
beta[ob]:=0;
for ta:=1 to ncomp do coef[ob,ta]:=0;
end;
write('      Beta ',ob:2,' REBUTJADA ');
end;

procedure complex(j8:integer);

```

```

var i8,i7,sv,tv,s,v,final:integer;
su,fun,rese:array[1..tcomp] of real;
caj,jac:sixpsix;
pere,prod,hc,otro,sn,kas:real;
begin
qs[j8]:=0;
if flag5='V' then final:=ncomp-1 else final:=ncomp;
for i8:=first[j8] to last[j8] do begin
su[ncomp]:=h^[j8,i8];
if switchcon then begin
for i7:=1 to (ncomp-1) do su[i7]:=cl^[j8,i8,i7];
end else begin
if i8=1 then begin
kas:=su[ncomp];
for v:=1 to final do su[v]:=prime(j8,v,kas);
end else
for i7:=1 to (ncomp-1) do su[i7]:=cl^[j8,i8-1,i7];
end;
repeat
for sv:=1 to ncomp do begin
fun[sv]:=0;
for tv:=1 to ncomp do begin
jac[sv,tv]:=0;
caj[sv,tv]:=0;
end;
end;
for s:=1 to nesp do begin
prod:=alog(beta[s]);
for sv:=1 to ncomp do prod:=prod*RS(su[sv],coef[s,sv]);
for v:=1 to ncomp do begin
fun[v]:=fun[v]+prod*coef[s,v];
for sv:=1 to final do
jac[v,sv]:=jac[v,sv]+prod*coef[s,v]*coef[s,sv];
end;
end;
for s:=1 to ncomp do begin
fun[s]:=fun[s]+su[s];
jac[s,s]:=jac[s,s]+su[s];
end;
hc:=kw/su[ncomp];
fun[ncomp]:=fun[ncomp]-hc;
jac[ncomp,ncomp]:=jac[ncomp,ncomp]+hc;
pere:=0;
for s:=1 to final do begin
fun[s]:=fun[s]-tot^[j8,i8,s];

```

```

pere:=pere+abs(fun[s]);
cl^[j8,i8,s]:=su[s];
end;
matinv(jac,final,caj);
for v:=1 to final do begin
rese[v]:=0;
for s:=1 to final do begin
rese[v]:=rese[v]+fun[s]*caj[s,v];
end;
end;
for v:=1 to final do begin
otro:=rese[v];
if abs(otro)>3 then otro:=signo(otro)*1.6;
su[v]:=su[v]/exp(otro);
end;
until pere<1.0E-9;
if flag5='V' then begin
volcor^[j8,i8]:=((tho[j8]-fun[ncomp])*vo[j8])/((fun[ncomp]-ht[j8]));
dif^[j8,i8]:=volume^[j8,i8]-volcor^[j8,i8];
end else begin
phcor^[j8,i8]:=-lgt(cl^[j8,i8,ncomp]);
dif^[j8,i8]:=ph^[j8,i8]-phcor^[j8,i8];
end;
qs[j8]:=qs[j8]+sqr(dif^[j8,i8]);
end;
end;
end;

```

```

procedure morespecies;
var im:integer;
add:char;
begin
writeln('          vols Afegir o Treure especies ?');
repeat
read(add) until add in['A','T'];
if add='A' then begin
nosp:=nosp+1;
for im:=1 to ncomp do begin writeln;
write('          Escriu el coeficient del component no. ',
im:2, '==>');
readln(coef[nosp,im]);
writeln;
end;
write('          Posa el log(Beta) aproximad      ==>');
readln(beta[nosp]);
writeln;

```

```

param[nesp]:=beta[nesp];
end      else begin
write('  Quina especie vols rebutjar ? ==>');
readln(im);
writeln;
outbeta(im);
writeln;
end;
end;

procedure where(kr:integer);
begin
case flag1 of
'G':gran(kr);
'P':pkas(kr);
'C':complex(kr);
end;
end;

procedure changerefine;
var jk:integer;
begin writeln;
writeln('  vols refinar per Volum o per p[H]');
writeln('      ^                ^ ');
write('      (ara refina per ');
case flag5 of
'V':writeln('volum)');
'H':writeln('p[H])');
end;
repeat
read(flag5);
until flag5 in ['V','H'];
for jk:=1 to ntit do where(jk);
end;

procedure changeph(j6:integer);
const Z1=0.434294;
var i6:integer;
fu,fd,h1,h2:real;
begin
for i6:=first[j6] to last[j6] do begin
h1:=alog((emf^[j6,i6]-eo[j6])/sl[j6]);
repeat
fu:=eo[j6]+sl[j6]*lgt(h1)+jh*h1+joh*kw/h1-emf^[j6,i6];
fd:=sl[j6]*z1/h1+jh-joh*kw/sqr(h1);

```

```

h2:=h1;
h1:=ABS(h2-fu/fd);
until abs(fu)<0.01;
h^[j6,i6]:=h2;
ph^[j6,i6]:=-lgt(h2);
end;
end;

```

```

procedure changeparam(sw1:integer);
var k2,number1,number2:integer;
begin
number1:=sw1 DIV 10;
number2:=sw1 MOD 10;
if sw1<=20 then begin
if flag1='G' then begin pkw:=param[1];
kw:=1/alog(pkw);
jh:=param[2];
joh:=param[3];
for k2:=1 to ntit do changeph(k2);
end else beta[sw1]:=param[sw1];
end else begin
case number1 of
2:begin eo[number2]:=param[sw1];changeph(number2);end;
3:begin sl[number2]:=param[sw1];changeph(number2);end;
4:begin ht[number2]:=sign[number2]*alog(param[sw1]);
setconc(number2);end;
5..8:begin co[number2,number1-4]:=alog(param[sw1]);
setconc(number2);end;
end;
end;
end;

```

```

procedure print;
var a1,b1,ip,tn:integer;
jt:string[13];
cus:string[2];
begin
writeln;
jt:= ' ';
for a1:=1 to ntit do begin
writeln(jt,jt,chr(129), ' ',fname[a1], ' ');
writeln(chr(128));
write(jt,chr(129), 'Vo=',vo[a1]:3:2, ' mV', ' ', 'Eo=',
eo[a1]:4:2, ' mV', ' ', 'g=',sl[a1]:2:3);
writeln(' Ho=',mil*co[a1,ncomp]:3:3, ' mM', ' Ht=',

```

```

mil*ht[a1]:3:3,' mM ');
write(chr(128));
if flag1<>'G' then begin
write(jt,chr(129),' Ao=',mil*co[a1,1]:3:3,' mM');
if ncomp>2 then write('   Bo=',mil*co[a1,2]:3:3,' mM ');
if ncomp>3 then write('   Co=',mil*co[a1,3]:3:3,' mM ');
end;
writeln;
writeln(chr(128));
write(jt,chr(133),' Vt(ml)          E(mV)          p[H]exp      ');
if flag5='V' then write(' Vtcalc          ') else
write(' p[H]calc          ');
writeln('error ');
writeln(chr(128));
for b1:=first[a1] to last[a1] do begin
if volume^[a1,b1]<10 then cus:=' ' else cus:=' ';
write(jt,chr(137),cus,volume^[a1,b1]:3:3,emf^[a1,b1]:13:1,
ph^[a1,b1]:13:3);
if flag5='V' then
writeln(volcor^[a1,b1]:12:3,dif^[a1,b1]:12:3,' ')
else writeln(phcor^[a1,b1]:12:3,dif^[a1,b1]:12:3,' ');
write(chr(128));
end;
writeln;
end;
qres:=residuals;
stdev:=stv;
writeln;
writeln(jt,jt,chr(137),' Suma de quadrats de residuals = ',
qres:9,' ');
write(chr(128));
writeln(jt,jt,chr(137),'          Desviació standard = ',
stdev:9,' ');
writeln(chr(128));
if flag1='G' then
writeln('          pKw= ',pkw:2:3,'   jh= ',jh:2:1,'   joh= ',joh:2:1)
else if flag1='P' then begin for tn:=1 to nesp do begin
writeln('          Beta',tn:1,' = ',beta[tn]:2:3);
end;
end else begin
for tn:=1 to nesp do begin
write('          log Beta');
for ip:=1 to ncomp do write(coef[tn,ip]:2);
writeln(' = ',beta[tn]:3:3);
end;
end;

```

```

end;
writeln;
end;

procedure printout;
var a1,b1,ip,tn:integer;
jt:string[5];
car:char;
cus:string[2];
begin
jt:= '    ';
for a1:=1 to ntit do begin
writeln(listing,jt,jt,' ',fname[a1],' ');
writeln(listing );
write(listing,jt,'Vo=',vol[a1]:3:2,' ml',          ' Eo=',
eo[a1]:4:2,' mV', ' g=',sl[a1]:2:3);
writeln(listing,          ' Ho=',mil*co[a1,ncomp]:3:3,' mM',
'Ht=',mil*ht[a1]:3:3,' mM ');
if flag1<>'G' then begin
write(listing,jt,' Ao=',mil*co[a1,1]:3:3,' mM');
if ncomp>2 then
write(listing,'      Bo=',mil*co[a1,2]:3:3,' mM ');
if ncomp>3 then
write(listing,'      Co=',mil*co[a1,3]:3:3,' mM ');
end;
writeln( listing);
writeln(listing);
write(listing,jt,' Vt(ml)          E(mV)          p[H]exp          ');
if flag5='V' then write(listing,' Vtcalc          ') else
write(listing,' p[H]calc          ');
writeln(listing,'error ');
writeln(listing,jt,
'-----');
for b1:=first[a1] to last[a1] do begin
if volume^[a1,b1]<10 then cus:=' ' else cus:=' ';
write(listing,jt,cus,volume^[a1,b1]:3:3,emf^[a1,b1]:13:1,
ph^[a1,b1]:13:3);
if flag5='V' then writeln(listing,volcor^[a1,b1]:12:3,
dif^[a1,b1]:12:3,' ') else
writeln(listing,phcor^[a1,b1]:12:3,dif^[a1,b1]:12:3,' ');
write(chr(128));
end;
writeln(listing);
end;
qres:=residuals;

```

```

stdev:=stv;
writeln(listing);
writeln(listing,jt,jt,' Suma de quadrats de residuals = ',
qres:9,' ');
writeln(listing,jt,jt,'          Desviació estandard = ',
stdev:9,' ');
writeln(listing );
if flag1='G' then writeln(listing,'          pKw= ',pkw:2:3,
'jh= ',jh:2:1,' joh= ',joh:2:1) else
if flag1='P' then begin
for tn:=1 to nesp do begin
writeln(listing,'          log Beta',tn:1,' = ', beta[tn]:2:3);
end;
end else begin
for tn:=1 to nesp do begin write(listing,'          log Beta');
for ip:=1 to ncomp do write(listing,coef[tn,ip]:2);
writeln(listing,' = ',beta[tn]:3:3);
end;
end;
writeln(listing);
end;

procedure changepoints;
var  il,jl:integer;
begin
total:=0;
for il:=1 to ntit do begin
writeln('          ',npoints[i1], ' punts ');
write('          Eo=',eo[i1]:4:2,' mV', ' g=',sl[i1]:2:3);
writeln(' Ho=',mil*co[i1,ncomp]:3:3,' mM', ' Ht=',
mil*ht[i1]:3:3,' mM');
if ncomp>1 then
write('          Ao=',mil*co[i1,1]:3:3,' mM');
if ncomp>2 then
write(' Bo=',mil*co[i1,2]:3:3,' mM');
if ncomp>3 then
write(' Co=',mil*co[i1,3]:3:3,' mM');
writeln;
writeln;
writeln('          punt          volum(ml)          emf(mV)');
writeln('          -----');
for jl:=1 to npoints[i1] do begin
writeln(jl:10,volume^[i1,jl]:21:3,emf^[i1,jl]:21:1);
end;
writeln;

```

```

writeln('      Primer i últim punt de la valoració no. ',il:1,'?');
readln(first[i1],last[i1]);
writeln;
total:=total+last[i1]-first[i1]+1;
setconc(i1);
changept(i1);
where(i1);
end;
npref:=0;
print;
end;

procedure muestra;
var l,m:integer;
begin
writeln;
if flag1='G' then
writeln('      1:pKw      2: (jh)      3: (joh)');
if flag1='P' then begin
for l:=1 to nesp do
writeln('      param. ', l:2,': log(Beta',l:2,')');
end;
if flag1='C' then begin
for l:=1 to nesp do begin
write('      param.',l:2,': log(Beta');
for m:=1 to ncomp do write(coef[l,m]:2);
writeln(')');
end;
end;
writeln;
write('      2x: Eo      3x: g      4x: Ht');
case ncomp of
1:writeln('      5x: Ho');
2:writeln('      5x: Ao      6x: Ho');
3:writeln('      5x: Ao      6x: Bo      7x: Ho');
4:writeln('      5x: Ao      6x: Bo      7x: Co      8x: Ho');
end;
writeln('      x: número de la valoració en particular');
writeln;
writeln;
end;

procedure calcula;
var pet:integer;
begin

```

```

for pet:=1 to ntit do where(pet);
end;

procedure refine;
var num1,num2,stp,r1,l,m,n,orden,np,k,alo,ik,niter,pre,il,
ola:integer;
arpar:array[1..mxt] of integer;
po,desvia:array[1..mxt] of real;
converge,finish1,finish,nosuccess,dib:boolean;
rner,enrr:sixpsix;
incr,Ma,kaput,ma2:real;
ind:array[1..mxt] of boolean;
begin
Ma:=0;
niter:=0;
writeln(' Quants parametres vols refinar ? (max.:',mxt:1,')');
readln(npref);
writeln('          tens ',npref:2,
' parámetros per refinar');
writeln;
muestra;
for r1:=1 to npref do begin
write('          Entra el número del parámetro ==>');
readln(arpar[r1]);
po[r1]:=param[arpar[r1]];
writeln;
end;
qres0:=residuals;
for il:=1 to npref do ind[i1]:=true ;
repeat
converge:=false;
finish:=false;
finish1:=false;
nosuccess:=false;
dib:=true;
stp:=0;
niter:=niter+1;
if niter>15 then nosuccess:=true;
if selector then begin
pre:=0;
for il:=1 to npref do begin
if ind[i1] then begin
pre:=pre+1;
arpar[pre]:=arpar[i1];
po[pre]:=po[i1];

```

```

ind[pre]:=true;
end;
end;
npref:=pre;
end;
for l:=1 to ntit do begin
for n:=first[l] to
last[l] do begin
stp:=stp+1;
error^[stp]:=dif^[l,n];
secdif^[l,n]:=dif^[l,n];
end;
end;
for np:=1 to npref do begin
orden:=arpar[np];
if param[orden]<>0 then incr:=0.0000001*param[orden]
else incr:=0.01;
param[orden]:=param[orden]+incr;
num1:=orden DIV 10;
num2:=orden MOD 10;
changeparam(orden);
stp:=0;
for k:=1 to ntit do begin
if (orden<=20) or (k=num2) then where(k);
for n:=first[k] to last[k] do begin
stp:=stp+1;
deriv^[np,stp]:=(secdif^[k,n]-dif^[k,n])/incr;;
end;
end;
param[orden]:=param[orden]-incr;
changeparam(orden);
end;
for l:=1 to npref do begin
for m:=1 to npref do begin
rner[l,m]:=0;
for n:=1 to total do begin
rner[l,m]:=rner[l,m]+deriv^[l,n]*deriv^[m,n] ;
end;
end;
end;
matinv(rner,npref,enrr);
for l:=1 to npref do begin
sh[l]:=0;
for m:=1 to total do begin
sh[l]:=sh[l]+deriv^[l,m]*error^[m];

```

```

end;
end;
for l:=1 to npref do begin
shift[l]:=0;
for m:=1 to npref do begin
shift[l]:=shift[l]+enrr[m,l]*sh[m];
end;
end;
writeln;
writeln('                Iteració ',niter:2);
writeln('          Parametres          Increments');
writeln('          -----          -----');
for l:=1 to npref do begin
alo:=arpar[l];
desvia[l]:=stv*sqrt(abs(enrr[l,l]));
if (flag1<>'G') and (alo<=20) then begin
if abs(shift[l])>2 then shift[l]:=2*signo(shift[l]);
end;
writeln('          ',alo:3,'          ',po[l]:8:3,shift[l]:20:4);
param[alo]:=po[l]+shift[l];
changeparam(alo);
end;
calcula;
writeln;
if selector then begin
for ik:=1 to npref do begin
kaput:=abs(desvia[ik]);
ola:=arpar[ik];
if (ola<=20) and (kaput>0.5*abs(param[ola]))
then begin
ind[ik]:=false;
outbeta(ola);
writeln('en la iteració ',niter:2);
dib:=false;
end;
end;
if not(dib) then begin
calcula;
qres0:=residuals;
end;
end;
if dib then begin qres4:=residuals;
writeln;
writeln(' Suma de quadrats de residuals=',qres4);
qres5:=qres4-qres0;

```

```

if (qres5<=0) then converge:=true ;
if converge then
for l:=1 to npref do po[l]:=param[arpar[l]]
else begin
qres1:=qres0;
qres3:=qres4;
for k:=1 to npref do begin
alo:=arpar[k];
if ind[k] then param[alo]:=po[k]+0.5*shift[k] else param[alo]:=0;
changeparam(alo);
end;
calcula;
qres2:=residuals;
Ma:=(qres1-2*qres2+qres3);
if Ma<>0 then Ma:=0.5+0.25*(qres1-qres3)/Ma else Ma:=0.99;
writeln('      Divergent: parametre de Wenworth = ',Ma);
for k:=1 to npref do begin
alo:=arpar[k];
param[alo]:=po[k]+Ma*shift[k];
changeparam(alo);
end;
calcula;
qres4:=residuals;
qres5:=qres4-qres0;
writeln('      Umin= ',qres0,' U= ',qres4);
if (qres5<=0) then begin
for l:=1 to npref do begin
alo:=arpar[l];
po[l]:=param[alo];
converge:=true;
end;
end else nosuccess:=true;
end;
if abs(qres5/qres0)<0.001 then finish1:=true;
if converge then qres0:=qres4;
if (converge) and (finish1) then finish:=true;
end;
until (finish) or (nosuccess) ;
if nosuccess then begin
for l:=1 to npref do begin
alo:=arpar[l];
param[alo]:=po[l];
changeparam(alo);
end;
calcula;

```

```

end;
if (finish) or (nosuccess) then begin
print;
if nosuccess then begin
write('      COMPTE : EL REFINAMENT NO HA CONVERGIT ');
if niter>15 then
writeln('      EN 15 ITERACIONS')      else
writeln('AL FINAL');
end;
writeln('      Parámetros refinats en ',
niter:2,'iteracions:');
writeln('      -----');
for l:=1 to npref do begin
alo:=arpar[l];
write('      Parámetro ',alo:2,' : ',param[alo]:3:3,' +/- ');
writeln(desvia[l]:2:4);
end;
end;
end;

procedure cuadr(x,y:real);
begin
move(x-ix,y-iy);
line(x-ix,y+iy);
line(x+ix,y+iy);
line(x+ix,y-iy);
line(x-ix,y-iy);
end;

procedure circle(x,y:real;angle,fl:integer);
var x1,y1,deg,lx,ly:real;
dgree,zl:integer;
begin if fl=0 then begin
for dgree:=1 to angle+1 do begin
deg:=dgree*2*pi/angle;
x1:=ix*sin(deg);
y1:=iy*cos(deg);
if dgree=1 then move(x+x1,y+y1)
else line(x+x1,y+y1);
end;
end else begin
for zl:=1 to 4 do begin
for dgree:=1 to angle+1 do begin
deg:=dgree*2*pi/angle;
x1:=zl*ix*sin(deg)/5;

```

```

y1:=z1*iy*cos(deg)/5;
if dgree=1 then move(x+x1,y+y1) else line(x+x1,y+y1);
end;
end;
end;
end;

```

```

procedure plot(xg,yg:real;wt:integer);
begin case wt of
1:cuadr(xg,yg);      {cuadrado}
2:circle(xg,yg,3,0); {triángulo}
3:circle(xg,yg,4,0); {rombo}
4:circle(xg,yg,6,0); {hexágono}
5:circle(xg,yg,25,0);{círculo}
6:circle(xg,yg,3,1); {triángulo negro}
7:circle(xg,yg,4,1); { rombo " }
8:circle(xg,yg,6,1); {hexágono " }
9:circle(xg,yg,25,1);{círculo " }
end;
end;

```

```

procedure titcurve;
var i,gt,gp,lab,k1,k2,k3,cnt:integer;
xmax,xmin,ymax,ymin,pmax,pmin,inx,iny,wx,wy:real;
sg:string[4] ;
begin
pmax:=phmax;
pmin:=phmin;
inx:=(volmax-volmin)/10;
iny:=(pmax-pmin)/10;
xmin:=volmin-1.5*inx;
xmax:=volmax+0.5*inx;
ymin:=pmin-iny;
ymax:=pmax+iny;
sx:=(xmax-xmin)/200;
sy:=(ymax-ymin)/150;
set_window(xmin,xmax,ymin,ymax);
wx:=3*sx;
wy:=5*sy;
set_char_size(wx,wy);
ix:=sx*factor;
iy:=sy*factor;
setstrlen(sg,2);
for lab:=0 to 25 do begin
if (lab<=volmax) and (lab>=volmin) then begin

```

```

move(lab,pmin+sy);
line(lab,pmin);
move(lab-wx/3,pmin-6*sy);
if lab<10 then setstrlen(sg,1) else setstrlen(sg,2);
strwrite(sg,1,cnt,lab:1);
gtext(sg);
end;
end;
move(volmin+7.5*inx,pmin+4*sy);
gtext('ml afegits');
for lab:=0 to 14 do begin
if (lab>=pmin) and (lab<=pmax) then begin
move(sx+volmin,lab);
line(volmin,lab);
move(volmin-6*sx,lab-wy/3);
if lab<10 then setstrlen(sg,1) else setstrlen(sg,2);
strwrite(sg,1,cnt,lab:1);
gtext(sg);
end;
end;
move(5*sx+volmin,pmax-6*sy);
gtext('-log[H]');
move(volmin,pmin);
line(volmax,pmin);
line(volmax,pmax);
line(volmin,pmax);
line(volmin,pmin);
for gt:=1 to ntit do begin
for gp:=first[gt] to last[gt] do begin
inx:=volume^[gt,gp];
iny:=ph^[gt,gp];
plot(inx,iny,gt);
end;
end;
for gt:=1 to ntit do begin
if flag5='V' then
move(volcor^[gt,first[gt]],ph^[gt,first[gt]]) else
move(volume^[gt,first[gt]],phcor^[gt,first[gt]]);
for gp:=first[gt] to last[gt] do begin
if flag5='V' then line(volcor^[gt,gp],ph^[gt,gp]) else
line(volume^[gt,gp],phcor^[gt,gp]);
end;
end;
end;

```

```

y1:=z1*iy*cos(deg)/5;
if dgree=1 then move(x+x1,y+y1) else line(x+x1,y+y1);
end;
end;
end;
end;

```

```

procedure plot(xg,yg:real;wt:integer);
begin case wt of
1:cuadr(xg,yg);      {cuadrado}
2:circle(xg,yg,3,0); {triángulo}
3:circle(xg,yg,4,0); {rombo}
4:circle(xg,yg,6,0); {hexágono}
5:circle(xg,yg,25,0);{círculo}
6:circle(xg,yg,3,1); {triángulo negro}
7:circle(xg,yg,4,1); { rombo      "  }
8:circle(xg,yg,6,1); {hexágono   "  }
9:circle(xg,yg,25,1);{círculo    "  }
end;
end;

```

```

procedure titcurve;
var i,gt,gp,lab,k1,k2,k3,cnt:integer;
xmax,xmin,ymax,ymin,pmax,pmin,inx,iny,wx,wy:real;
sg:string[4] ;
begin
pmax:=phmax;
pmin:=phmin;
inx:=(volmax-volmin)/10;
iny:=(pmax-pmin)/10;
xmin:=volmin-1.5*inx;
xmax:=volmax+0.5*inx;
ymin:=pmin-iny;
ymax:=pmax+iny;
sx:=(xmax-xmin)/200;
sy:=(ymax-ymin)/150;
set_window(xmin,xmax,ymin,ymax);
wx:=3*sx;
wy:=5*sy;
set_char_size(wx,wy);
ix:=sx*factor;
iy:=sy*factor;
setstrlen(sg,2);
for lab:=0 to 25 do begin
if (lab<=volmax) and (lab>=volmin) then begin

```

```

move(lab,pmin+sy);
line(lab,pmin);
move(lab-wx/3,pmin-6*sy);
if lab<10 then setstrlen(sg,1) else setstrlen(sg,2);
strwrite(sg,1,cnt,lab:1);
gtext(sg);
end;
end;
move(volmin+7.5*inx,pmin+4*sy);
gtext('ml afegits');
for lab:=0 to 14 do begin
if (lab>=pmin) and (lab<=pmax) then begin
move(sx+volmin,lab);
line(volmin,lab);
move(volmin-6*sx,lab-wy/3);
if lab<10 then setstrlen(sg,1) else setstrlen(sg,2);
strwrite(sg,1,cnt,lab:1);
gtext(sg);
end;
end;
move(5*sx+volmin,pmax-6*sy);
gtext('-log[H]');
move(volmin,pmin);
line(volmax,pmin);
line(volmax,pmax);
line(volmin,pmax);
line(volmin,pmin);
for gt:=1 to ntit do begin
for gp:=first[gt] to last[gt] do begin
inx:=volume^[gt,gp];
iny:=ph^[gt,gp];
plot(inx,iny,gt);
end;
end;
for gt:=1 to ntit do begin
if flag5='V' then
move(volcor^[gt,first[gt]],ph^[gt,first[gt]]) else
move(volume^[gt,first[gt]],phcor^[gt,first[gt]]);
for gp:=first[gt] to last[gt] do begin
if flag5='V' then line(volcor^[gt,gp],ph^[gt,gp]) else
line(volume^[gt,gp],phcor^[gt,gp]);
end;
end;
end;
end;

```

```

procedure distr;
const maxlen=5;
var t,tt,tp,asd,t1,t2,dnt,t3:integer;
xmax,xmin,ymax,ymin,dmax,dmin,inx,iny,wx,wy,ct,cl:real;
dg,de,as:string[7];
begin
inx:=(phmax-phmin)/7;
iny:=6*stv/7;
dmax:=3*stv;
dmin:=-dmax;
xmax:=phmax+inx;
xmin:=phmin-inx;
ymin:=dmin-iny;
ymax:=dmax+iny;
sx:=(xmax-xmin)/200;
sy:=(ymax-ymin)/150;
set_window(xmin,xmax,ymin,ymax);
wx:=3*sx;
wy:=5*sy;
set_char_size(wx,wy);
ix:=sx*factor;
iy:=sy*factor;
for t:=1 to 14 do begin
if t<10 then setstrlen(dg,1) else setstrlen(dg,2);
if (t>=phmin) and (t<=phmax) then begin
move(t,dmin+sy);
line(t,dmin);
move(t-wx/3,dmin-6*sy);
strwrite(dg,1,dnt,t:1);
gtext(dg);
end;
end;
ct:=-lgt(dmax/2);
if ct>2.5 then cl:=0.001;
if (2.5>=ct) and (ct>2) then cl:=0.005;
if (2>=ct) and (ct>1.5) then cl:=0.01;
if 1.5>=ct then cl:=0.05;
setstrlen(as,1);
for t:=-8 to 8 do begin
if t=0 then begin move(phmin-18*sx,-sy/3);
gtext(' 0.00');
end else begin
ct:=t*cl;
if (ct>=dmin) and (ct<=dmax) then begin
move(phmin+sx,ct);

```

```

line(phmin,ct);
if t<0 then as:='- ' else as:=' ';
if abs(c1)<0.01 then begin
setstrlen(de,maxlen);
strwrite(de,1,dnt,abs(ct):1:3);
end else begin
setstrlen(de,maxlen-1);
strwrite(de,1,dnt,abs(ct):1:2);
end;
move(phmin-18*sx,ct-sy/3);
gtext(as);
gtext(de);
end;
end;
end;
move(phmin,dmin);
line(phmax,dmin);
line(phmax,dmax);
line(phmin,dmax);
line(phmin,dmin);
move(phmin,0);
line(phmax,0);
set_line_style(7);
move(phmin,2*stv);
line(phmax,2*stv);
set_line_style(1);
gtext(' 2 s.d. ');
set_line_style(7);
move(phmin,stv);
line(phmax,stv);
set_line_style(1);
gtext(' s.d. ');
set_line_style(7);
move(phmin,-stv);
line(phmax,-stv);
set_line_style(1);
gtext('- s.d. ');
set_line_style(7);
move(phmin,-2*stv);
line(phmax,-2*stv);
set_line_style(1);
gtext('-2 s.d. ');
move(phmin-3*sx,dmax+3*sy);
gtext('error');
move(phmax+2*sx,dmin);

```

```

gtext('-log[H]');
for tt:=1 to ntit do begin
for tp:=first[tt] to last[tt] do
begin inx:=ph^[tt,tp];
iny:=dif^[tt,tp];
plot(inx,iny,tt);
end;
end;
end;

procedure grafic;
const control=0;
var gdu,a,b:char;
gda,err,gft,gfp:integer;
begin
writeln;
writeln('          Curva_de_valoració o Distribució de residuals');
writeln('          ^                               ^');
repeat read(b) until b in ['C','D'];
writeln;
writeln('          Plotter o pAntalla');
writeln('          ^             ^');
repeat read(gdu) until gdu in ['P','A'];
case gdu of
'P': begin factor:=0.7;
gda:=705;
end;
'A': begin factor:=1;
gda:=3;
page;
end;
end;
phmin:=14;
phmax:=0;
volmax:=0;
volmin:=30;
for gft:=1 to ntit do begin
if volume^[gft,last[gft]]>=volmax then
volmax:=volume^[gft,last[gft]];
if volume^[gft,first[gft]]<=volmin then
volmin:=volume^[gft,first[gft]];
for gfp:=first[gft] to last[gft] do begin
if ph^[gft,gfp]<=phmin then phmin:=ph^[gft,gfp];
if ph^[gft,gfp]>=phmax then phmax:=ph^[gft,gfp];
end;

```

```

end;
phmin:=trunc(phmin);
phmax:=trunc(phmax+1);
volmin:=trunc(volmin);
volmax:=trunc(volmax+1);
graphics_init;
display_init(gda,control,err);
set_aspect(2,1.5);
case b of
'C':titcurve;
'D':distr;
end;
writeln;
writeln('          ',chr(131),
 '< prem qualsevol tecla per continuar >');
read(a);
write(chr(128));
page;
clear_display;
graphics_term;
end;

procedure manual;
var man1,man2,man3,man4:integer;
man5:real;
begin
muestra;
write('      Entra el número del parametre per canviar ==>');
readln(man1);
writeln;
write('      Entra el nou valor del parametre ',man1:2, ' ==>');
readln(man5);
writeln;
if man1>40 then man5:=lgt(abs(man5));
param[man1]:=man5;
changeparam(man1);
man3:=man1 MOD 10;
for man4:=1 to ntit do begin
if (man1<=20) or (man3=man4) then where(man4);
end;
page;
end;

procedure menu;
var mu:integer;

```

```

begin writeln;
write(' Ajustar_parametres');
if flag1='C' then begin if selector then
write(' seLector(si)') else write(' seLector(no)');
writeln(' canviar_reFinament  proVar_especies');
END;
writeln;
writeln(
'      Refinar  Canviar_punts  Grafics  imPrimir
veure_reSultats  fI');
repeat  read(flag2)
until flag2 in ['F','R','C','G','P','A','S','V','I','L'];
case flag2 of
'V':if com then begin morespecies;for mu:=1 to ntit do where(mu);
end;
'F':if com then changerefine;
'R':refine;
'C': changepoints;
'A':manual;
'P':printout;
'L': if com then if selector then selector:=false else
selector:=true;
'G':grafic;
'S':print;
'I':begin page;
acabado:=true;
end;
end;
end;
end;

```

```

begin
new(volume);
new(emf);
new(dif);
new(ph);
new(phcor);
new(volcor);
new(tot);
new(cl);
new(h);
new(deriv);
new(error);
new(secdif);

```

```

selector:=false;
com:=false;
writeln(' Tipus de valoració: Gran Pkas Complexes ?');
writeln('      ^      ^      ^');
repeat read(flag1) until flag1 in ['G','P','C'];
writeln;
writeln('      Quin és el nom del model proposat ?');
readln(fmodel);
reset(model,fmodel+'.TEXT');
if flag1='G' then begin pl[1]:=1;
ncomp:=1;
read(model,pkw,jh,joh);
param[1]:=pkw;
param[2]:=jh;
param[3]:=joh;
end;
if flag1='P' then begin
ncomp:=2;
pl[2]:=1;
read(model,pkw,jh,joh,pl[1],nesp);
for i:=1 to nesp do begin read(model,beta[i]);
param[i]:=beta[i];
end;
end;
if flag1='C' then begin
com:=true;
read(model,pkw,jh,joh,ncomp,nesp);
for i:=1 to ncomp do read(model,pl[i]);
for i:=1 to nesp do begin
for j:=1 to ncomp do read(model,coef[i,j]);
read(model,beta[i]);
param[i]:=beta[i];
end;
end;
kw:=1/alog(pkw);
writeln('      Quantes valoracions ?');
readln(ntit);
acabado:=false;
for i:=1 to ntit do begin
writeln('      Nom de la valoració no.',i:2);
readln(fname[i]);
reset(f,fname[i]);
read (f,npoints[i],vo[i],eo[i],sl[i],ht[i]);
param[20+i]:=eo[i];
param[30+i]:=sl[i];

```

```
param[40+i]:=lgt(abs(ht[i]));
sign[i]:=ht[i]/abs(ht[i]);
for j:=1 to ncomp do begin
read(f,co[i,j]);
if co[i,j]<>0 then param[40+10*j+i]:=lgt(abs(co[i,j]))
else param[40+10*j+i:=-100 ;
end;
if co[i,ncomp]<>0 then hsig[i]:=co[i,ncomp]/abs(co[i,ncomp])
else hsig[i]:=0;
for j:=1 to npoints[i] do read(f,volume^[i,j],emf^[i,j]);
end;
if flag1='C' then begin  flag5:='V';
switchcon:=false;
for i:=1 to ntit do begin
first[i]:=1;
last[i]:=npoints[i];
changept(i);
setconc(i);
where(i);
end;
switchcon:=true ;
end else flag5:='H';
changepoints;
repeat menu until acabado;
end.
```

