Parametric Programming Technique for Global Optimization of Wastewater Treatment Systems

João Teles,[a,b] Pedro Castro,[a] Henrique Matos[b]

[a]*U. Modelação Optimização Sistemas Energéticos, LNEG, 1649-038 Lisboa, Portugal,*
[b]*Dep. Engenharia Química e Biológica, IST, 1049-001, Lisboa, Portugal*

Abstract

This paper presents a parametric programming technique for the optimal design of industrial wastewater treatment networks (WTN) featuring multiple contaminants. Inspired in scientific notation and powers of ten, the proposed approach avoids the non-convex bilinear terms through a piecewise decomposition scheme that combines the generation of artificial flowrate variables with a multi-parameterization of the outlet concentration variables. The general non-linear problem (NLP) formulation is replaced by a mixed-integer linear programming (MILP) model that is able to generate near optimal solutions, fast. The performance of the new approach is compared to that of global optimization solver BARON through the solution a few test cases.

Keywords: Wastewater; Optimization; Mixed-Integer Linear Programming.

1. Introduction

Water is a resource that is used intensively for many different purposes in industry. Many of the processes are today subject to strict environmental regulations on discharge effluents due to increased water scarcity. Improved water management can effectively reduce freshwater demand and overall wastewater generation, and thus lower freshwater and effluent treatment costs.

Over the past decade, numerous research works have addressed this topic ranging from graphical pinch analysis techniques to mathematical optimization approaches. Graphical methods are easier to understand conceptually, while mathematical programming has a wider applicability scope. The mathematical programming approach relies commonly on the optimization of a superstructure for either integrated or separated problems. The optimal design of water-using networks (Teles et al., 2009), water treatment networks (Castro et al., 2009) or both integrated into one large system featuring regeneration and recycling (Gunaratnam et al., 2005; Karuppiah & Grossmann, 2006) can be formulated as (mixed-integer) non-linear programming problems (if logic constraints are used to prevent recycling). Such problems feature non-convex bilinear terms that make them very difficult to solve by gradient-based algorithms that are the basis of most commercial NLP solvers, which often cannot avoid getting trapped in suboptimal solutions. The complexity of non-convex NLP/MINLP problems is well documented in the literature and a number of different algorithms have been proposed for their solution such as: branch-and-bound, adaptive random-search, outer-approximation/equality relaxation, branch-and-reduce, generalized disjunctive programming, simulated annealing, MILP/LP heuristic search strategies, etc. While some approaches cannot ensure global optimality, others may require significant, if not prohibitively large, computational resources.

2. Problem statement

In this paper, we focus on the optimal design of a wastewater treatment system. Given a set of process wastewater streams *W* containing well-defined pollutants (set *C*) with known flowrates tf_w^{wwat} and concentrations $c_{w,c}^{wwat}$, the goal is to generate an effluent that meets discharge regulations c_c^{env} for all contaminants while minimizing the total flowrate going through the treatment units *T*. These are characterized by fixed removal ratios $rr_{i,c}$ and maximum inlet concentrations $c_{i,c}^{in\,max}$ ($i \in T$).

3. New parametric programming approach

To address this problem in a systematic way it is necessary to build a superstructure that embeds all possible flow configurations, similarly to any other optimization study in process synthesis: (i) each wastewater stream that enters the network can be sent to the treatment units or to the final discharge mixer (bypass); (ii) each unit is preceded by a mixer, which is fed by wastewater and reuse/recycle streams originating from the outlets of all treatment processes; (iii) each treatment unit is followed by a splitter that feeds the final discharge mixer, as well as other treatment processes; (iv) effluent streams from each unit and bypass streams are mixed in a final discharge mixer to ensure compliance with the environmental legislation.

To optimize these superstructures mathematical models are required and two alternative formulations have been proposed in the literature. They include non-convex bilinear terms in the mass balances of treatment units, involving either products of stream flowrates and concentrations in the mixers, or products of contaminant flowrates and split fractions in the splitters. Thus, if solved with local optimization solvers like CONOPT, suboptimal solutions are mostly likely to occur. To overcome this limitation, a new deterministic procedure is proposed. It is based on a piecewise decomposition scheme that approximates to a chosen accuracy level the non-convex terms in the original model. The resulting MILP generates a near optimal network that acts as an upper bound on the true global optimum. Furthermore, such output solution can easily be refined following initialization and solution of the general NLP with a local solver.

The required steps for converting the bilinear NLP into a MILP are described next.

3.1. Parameterization of the outlet concentration variables

Consider the well-known notation in the decimal numeral system that uses positions for each power of ten: units, tens, hundreds, thousands, etc., and ten different numerals, the digits 0, 1, 2, ..., 9 to represent any real number. It also requires a dot (decimal point) to represent decimal fractions. We follow this principle to approximate the outlet concentration for all contaminants in all treatment units. The dynamic construction process starts by fixing the number (n) of digits for the fractional part of the number (decimal region). The required number of digits ($k^{i,c}$) in the left-hand side of the decimal point (integral region) is then determined based on the maximum possible outlet concentration for contaminant *c* in unit *i*, which can be calculated from the given maximum inlet concentration and removal ratio. Thus, a total of $\xi_{i,c}=n+k^{i,c}$ digits are involved in the representation of the outlet concentration of contaminant *c* in unit *i*.

3.2. Disaggregation of the flowrate variables

The flowrate variables linked to the bilinear terms associated to the mass balances of unit *i*, give the flowrate from *i* to unit *j*, $F_{i,j}$, and the amount sent to the discharge mixer, F_i^{dis}. We now need to define, for each different contaminant *c*, multiple artificial variables resulting from the disaggregation of these base variables: $TF_{i,j,c}^{a,p}$ and $TD_{i,c}^{a,p}$, respectively. The two additional indices are critical, a identifies the numerical position ($a_{n+k,\ldots,}$ a_{n+1} . a_n, a_{n-1},…a_1) and p the ten different possible digits (0,…,9).

Fig. 1 illustrates the concept for variables $F_{i,j}$. The bilinear term $F_{i,j} \times C_{i,c}^{out}$, representing the contaminant mass in the stream linking the outlet splitter (circle) of unit i and the inlet mixer of unit j (diamond), is disaggregated into a sum of linear terms. Suppose that the optimal outlet concentration of contaminant c is equal to 1(…)0.8(…)3. If 6 digits (n=k=3) are used, the number can be generated by: 1E(+3)+…+0E0+8E(-1)+…+3E(-3). Notice that we are selecting a single digit per position, with the chosen values being identified through non-zero values of the decision variables $Z_{i,c}^{a,p}$. Then, only the corresponding artificial flowrate variables (highlighted in Fig. 1) can assume positive values so that the accurate component mass flow is generated. Moreover, all artificial variables $TF_{i,j,c}^{a,p}$ will be forced to have the same value, $F_{i,j}$.
Overall, the new parametric programming approach is versatile since the number of significant digits can be increased for more accurate optimal solutions and decreased for lower computational effort.

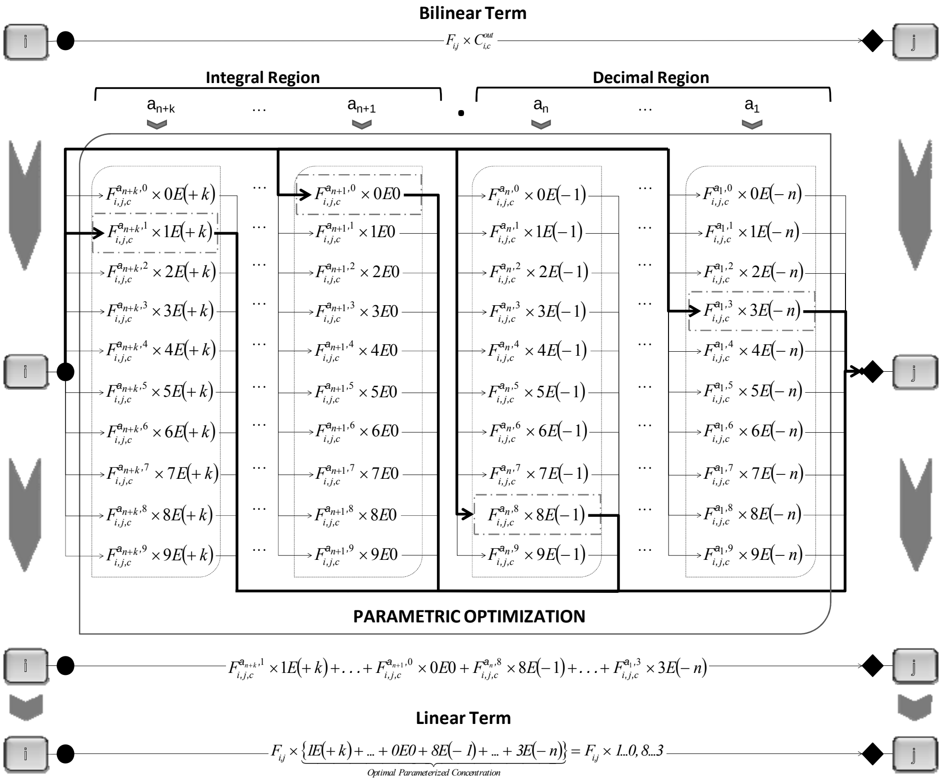

Figure 1. Illustration of parametric programming approach for the link between units i and j.

4. Mathematical formulation

The mixed integer linear programming formulation requires the additional positive continuous variables are required: F_i^{tot} gives the total flowrate entering/leaving treatment unit *i*; $F_{w,i}^{wwat}$ is the flowrate of wastewater *w* into treatment unit *i*; F_w^{byp} is the flowrate of wastewater *w* that bypasses the treatment system and goes directly to the final discharge mixer; $MT_{i,c}^{In}$ and $MT_{i,c}^{Out}$ are the inlet/outlet mass flows of contaminant *c* for treatment unit *i*.

The objective is to minimize the total flowrate, going through the treatment units, Eq. 1. Eq. 2 represents the flow balance over the splitters associated to the system's inlet wastewater streams. Eqs. 3-4 are the flowrate balances over the inlet mixer and outlet splitter linked to treatment unit *i*, while Eqs. 5-6 are the corresponding mass balances.

$$\min \sum_{i \in T} F_i^{tot} \tag{1}$$

$$tf_w^{wwat} = \sum_{i \in T} F_{w,i}^{wwat} + F_w^{byp}, \quad \forall w \in W \tag{2}$$

$$F_i^{tot} = \sum_{w \in W} F_{w,i}^{wwat} + \sum_{j \in T} F_{i,j}, \quad \forall i \in T \tag{3}$$

$$F_i^{tot} = \sum_{j \in T} F_{i,j} + F_i^{dis}, \quad \forall i \in T \tag{4}$$

$$MT_{i,c}^{In} = \sum_{w \in W} F_{w,i}^{wwat} \cdot c_{w,c}^{wwat} + \sum_{j \in T} \sum_{a=1}^{\xi_{i,c}} \sum_{p=0}^{9} TF_{j,i,c}^{a,p} \times \psi_{a,p}, \quad \forall i \in T, c \in C \tag{5}$$

$$MT_{i,c}^{Out} = \sum_{j \in T} \sum_{a=1}^{\xi_{i,c}} \sum_{p=0}^{9} TF_{i,j,c}^{a,p} \cdot \psi_{a,p} + \sum_{a=1}^{\xi_{i,c}} \sum_{p=0}^{9} TD_{i,c}^{a,p} \cdot \psi_{a,p}, \quad \forall i \in T, c \in C \tag{6}$$

Eqs. 7-8 ensure that all active artificial variables have the same value. Eq. 9 is the definition of the removal ratio of each contaminant within each unit. Eq. 10 ensures that the environmental discharge limits are not exceeded. Eq. 11 forces the artificial flowrate variables linked to non-selected parameters to be zero. On the other hand, the outlet flow cannot exceed a certain upper bound, where multiplicative factor φ is employed to allow for a recycled flow greater than the total amount of wastewater entering the treatment system. Eq. 12 ensures that a single digit is selected for a certain numerical position for contaminant c in unit i. Finally, Eqs. 13-14 guarantee that the inlet contaminants concentration does not exceed their maximum admissible values and that of outlet concentrations (LHS) are lower than the maximum outlet concentrations. Additional design constraints can be easily incorporated into the mathematical model.

$$F_i^{dis} = \sum_{p=0}^{9} TD_{i,c}^{a,p}, \quad \forall i \in T, c \in C, a \in \xi_{i,c} \tag{7}$$

$$F_{i,j} = \sum_{a=1} \sum_{p=0}^{9} TF_{i,j,c}^{a,p}, \quad \forall i,j \in T, c \in C, a \in \xi_{i,c} \tag{8}$$

$$MT_{i,c}^{Out} = MT_{i,c}^{In} \times \left(1 - rr_{i,c}\right), \quad \forall i \in T, c \in C \tag{9}$$

$$\sum_{w \in W} F_w^{byp} \cdot c_{w,c}^{wwat} + \sum_{i \in T} \sum_{a \in \xi_{i,c}} \sum_{p=0}^{9} TD_{i,c}^{a,p} \times \psi_{a,p} \le \left(\sum_{w \in W} F_w^{byp} + \sum_{i \in T} F_i^{dis} \right) \times c_c^{env}, \quad \forall c \in C \tag{10}$$

$$\sum_{j \in T} TF_{i,j,c}^{a,p} + TD_{i,c}^{a,p} \le \varphi \cdot \min \left[\sum_{w \in W} tf_w^{wwat}, \left(\sum_{w \in W} tf_w^{wwat} \times c_{w,c}^{wwat} \right) / c_{i,c}^{in\,\max} \right] \times Z_{i,c}^{a,p}, \tag{11}$$

$$\forall i \in T, c \in C, a \in \xi_{i,c}, p \in \{0,1,\ldots,9\}$$

$$\sum_{p=0}^{9} Z_{i,c}^{a,p} = 1, \quad \forall i \in T, c \in C, a \in \xi_{i,c} \tag{12}$$

$$C_{i,c}^{in} \le c_{i,c}^{in\,max}, \quad \forall i \in T, c \in C \tag{13}$$

$$\sum_{a=1}^{\xi_{i,c}} \sum_{p=0}^{9} \psi_{a,p} \times Z_{i,c}^{a,p} \le c_{i,c}^{out\,max}, \quad \forall i \in T, c \in C \tag{14}$$

5. Computational results

The performance of the new technique is now illustrated through several samples. Their respective sizes and numerical results are given in Figs. 2-3. The hardware consisted of an Intel Core 2 Duo 2.4 GHz processor, with 2 GB of RAM memory, running Windows Vista. The underlying formulations were implemented and solved in GAMS 23.2, using CPLEX as the MILP solver, and CONOPT and BARON as NLP solvers, the latter being a global optimization solver.

The results in Figure4 show that solution quality increases with an increase in the number of decimal places used to represent the outlet concentrations. This is true both for CPLEX and CONOPT, which is initialized with the former solution. Thus, it is not surprising to find out that CONOPT normally improves the solution, by making it more accurate. The highest deviation in the range [1, 4] occurs for Ex5, with the gap between the outputs from CPLEX and CONOPT being equal to 2.79, 0.08, 0.02 and 0%, respectively. For two decimal places (n=2) the gap is always under 0.10%, which is a very small number for all practical purposes. Furthermore, with the exception of Ex5 for n=1 and Ex11 for n=1-3, the solution from CONOPT is the global optimal solution. And only in the former the difference is relevant, 231.881 vs. 229.701.

BARON could also find the global optimal solution for all examples. However, the optimality gaps for Ex9-12 and Ex14 are very significant (roughly >10%), particularly after one takes into account the computational effort (CPUs in Fig. 3). In contrast, our new CPLEX+CONOPT approach took less than one hour for all problems but Ex14 for n=4. Despite the fact that we cannot calculate an optimality gap since there is no relaxation providing a lower bound as in BARON, it is fair to assume that the optimality gap will be lower than the difference between the output values from CPLEX for n=3 and n=4, which are below 0.02%.

Finally, it should be noted that there is not a linear relation between the number of decimals and computational effort. Frequently the CPUs decreased for increasing problem sizes (directly related to n), which can be explained by: (i) increasing n may lead to lower integrality gaps, which contributes to a faster search; (ii) the branch & bound algorithm beneath CPLEX is heuristic.

6. Conclusions

This work has presented a new strategy for the optimal design of wastewater treatment networks. It involves generating a set of artificial multi-parametric elements resulting from the decomposition of bilinear terms present in the general nonlinear mathematical formulation. The optimal elements are then chosen through the solution of a single MILP problem. The outcome is an upper bound on the global optimal solution that becomes increasingly tighter with an increase in the number of decimal digits used in the approximation. The same principle can be applied to other process synthesis problems. In particular, work is underway to extend the approach to the design of water-using networks and fully integrated water networks.

References

J. Teles, P. Castro, A. Novais, 2009, MILP-based initialization strategies for the optimal design of water-using networks. Chem. Eng. Sci. 64, 3736-3752.

P. Castro, J. Teles, A. Novais, 2009, Linear program-based algorithm for the optimal design of wastewater treatment systems. Clean Techn Environ Policy. 11, 83-93.

M. Gunaratnam, A. Alva-Argáez, A. Kokossis, J. Kim J & R. Smith, 2005, Automated design of total water systems. Ind. Eng. Chem. Res. 44, 588.

R. Karuppiah, I. Grossmann, 2006, Global optimization for the synthesis of integrated water systems in chemical processes. Comput. Chem. Eng. 30, 650.

	Number of decimal digits for optimal approximation of variable concentrations (n)														
Aprox	**1**			**2**			**3**			**4**			**BARON**		
Case Studies	**CPLEX**	**CONOPT**	Gap	**CPLEX**	**CONOPT**	Gap	**CPLEX**	**CONOPT**	Gap	**CPLEX**	**CONOPT**	Gap	Lower Bound	Sol.	Gap
Ex1	**179.798**	**179.798**	0.00%	**179.798**	**179.798**	0.00%	**179.798**	**179.798**	0.00%	**179.798**	**179.798**	0.00%	**179.798**	**179.798**	0.00%
Ex2	130.705	**130.703**	0.00%	130.705	**130.703**	0.00%	**130.703**	**130.703**	0.00%	**130.703**	**130.703**	0.00%	**130.703**	**130.703**	0.00%
Ex3	**99.495**	**99.495**	0.00%	**99.495**	**99.495**	0.00%	**99.495**	**99.495**	0.00%	**99.495**	**99.495**	0.00%	**99.495**	**99.495**	0.00%
Ex4	90.441	**89.836**	0.67%	89.929	**89.836**	0.10%	89.841	**89.836**	0.01%	89.837	**89.836**	0.00%	**89.836**	**89.836**	0.00%
Ex5	238.359	231.881	2.79%	229.849	**229.701**	0.06%	229.75	**229.701**	0.02%	229.708	**229.701**	0.00%	**229.701**	**229.701**	0.00%
Ex6	174.054	**173.478**	0.33%	173.529	**173.478**	0.03%	173.483	**173.478**	0.00%	173.479	**173.478**	0.00%	**173.478**	**173.478**	0.00%
Ex7	80.87	**80.779**	0.11%	80.783	**80.779**	0.00%	80.781	**80.779**	0.00%	80.78	**80.779**	0.00%	**80.779**	**80.779**	0.00%
Ex8	586.814	**586.68**	0.02%	586.702	**586.68**	0.00%	586.681	**586.68**	0.00%	586.681	**586.68**	0.00%	**586.68**	**586.68**	0.00%
Ex9	2127.168	**2127.115**	0.00%	2127.123	**2127.115**	0.00%	2127.116	**2127.115**	0.00%	2127.116	**2127.115**	0.00%	1534.628	**2127.115**	27.85%
Ex10	1201.038	**1201.038**	0.00%	**1201.038**	**1201.038**	0.00%	**1201.038**	**1201.038**	0.00%	**1201.038**	**1201.038**	0.00%	829.982	**1201.038**	30.89%
Ex11	1566.785	1564.958	0.12%	1564.992	1564.958	0.00%	1564.968	1564.958	0.00%	1564.959	**1564.957**	0.00%	1126.431	**1564.957**	28.02%
Ex12	513.875	**513.001**	0.17%	513.082	**513.001**	0.02%	513.054	**513.001**	0.01%	513.001	**513.001**	0.00%	404.629	**513.001**	21.13%
Ex13	2463.297	**2446.429**	0.69%	2447.114	**2446.429**	0.03%	2446.53	**2446.429**	0.00%	2446.43	**2446.429**	0.00%	**2446.429**	**2446.429**	0.00%
Ex14	1359.688	**1358.663**	0.08%	1358.792	**1358.663**	0.01%	1358.67	**1358.663**	0.00%	1358.668	**1358.663**	0.00%	1233.176	**1358.663**	9.24%

Figure 2. Solution quality as a function of the number of decimal digits (global solution in bold).

				SINGLE EQUATIONS				SINGLE VARIABLES				DISCRETE VARIABLES				CPUs				
				Number of decimal digits for optimal approximation of variable concentrations (n)												CPLEX + CONOPT				BARON
Case Studies	#C	#WS	#TU	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	
Ex1	1	3	1	37	50	63	76	72	102	132	162	20	30	40	50	0.14	0.07	0.07	0.07	0.05
Ex2	1	2	2	100	128	156	184	259	339	419	499	60	80	100	120	0.28	0.18	0.26	0.29	0.06
Ex3	1	3	2	87	115	143	171	222	302	382	462	50	70	90	110	0.18	0.18	0.19	0.18	0.08
Ex4	2	2	2	193	249	305	361	503	663	823	983	120	160	200	240	0.17	0.38	0.81	1.17	0.12
Ex5	3	3	3	514	649	784	919	1596	2046	2496	2946	310	400	490	580	38	111	130	245	10.95
Ex6	3	3	3	559	694	829	964	1746	2196	2646	3096	340	430	520	610	1068	274	347.97	421	40.26
Ex7	3	3	3	589	724	859	994	1846	2296	2746	3196	360	450	540	630	26	73	134.99	1335	303
Ex8	1	3	5	307	392	477	562	1184	1534	1884	2234	160	210	260	310	6	13	20	23	280
Ex9	1	5	7	486	619	752	885	2188	2818	3448	4078	230	300	370	440	6	11	11.056	27	8441[a]
Ex10	1	6	10	705	925	1145	1365	3676	4876	6076	7276	290	390	490	590	14	641	122	731	8295[a]
Ex11	1	6	15	1394	1799	2204	2609	8542	11092	13642	16192	480	630	780	930	3011	39	245	420	225780[a]
Ex12	2	3	5	583	753	923	1,093	2244	2944	3644	4344	310	410	510	610	2298	2927.5	722	2945	17262[a]
Ex13	3	4	3	469	604	739	874	1446	1896	2346	2796	280	370	460	550	153	112	435	1095	1254
Ex14	2	2	6	749	965	1181	1,397	3127	4087	5047	6007	380	500	620	740	131	1866.8	243	8028	78605[a]

[a]unable to prove global optimality (lower bound at time of interruption)

Figure 3. Computational statistics as a function of the number of decimal digits.