Hierarchical Quadratic Programming: Companion report
Adrien Escande, Nicolas Mansard, Pierre-Brice Wieber

To cite this version:
Adrien Escande, Nicolas Mansard, Pierre-Brice Wieber. Hierarchical Quadratic Programming: Companion report. 2013. <hal-00970816>

HAL Id: hal-00970816
https://hal.archives-ouvertes.fr/hal-00970816
Submitted on 2 Apr 2014

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Hierarchical Quadratic Programming:  
Companion report

Adrien Escande  
JRL-CNRS/AIST  
Tsukuba, Japan

Nicolas Mansard  
LAAS-CNRS, Univ. Toulouse  
Toulouse, France

Pierre-Brice Wieber  
INRIA Grenoble  
St Ismier, France

This technical report summarizes the technical content in addition to the main paper Escande et al. (2013). In the first section, we detail the complexity cost of the active-set algorithm. In Section 2, we recall the continuity property and gives the detailed proof. In Section 3, we recall the stability property of the inverse-kinematics (IK) control scheme using the hierarchical quadratic program (HQP) solver.

This report is not self-contained. It is to be read in addition to the main paper Escande et al. (2013). The same notations are used and, when not in the report, they are defined in the paper. All references from the papers are denoted by using the prefix π- (see Section π-XX or Equation (π-XX)).

1 Complexity

The active-set algorithm is composed of two distinctive parts. On the first hand, an equality-only HQP solver (eHQP) is computing the optimum for a given active set. The main computations are the HCOD decomposition and its inversion. See Section π-2 for details. On the second hand, the active-set loop searches the optimal active set by activating and deactivating one constraint of the active set. See Section π-3 for details. The activation and deactivation is selected based on the values of the primal and dual optima.
1.1 Equality-only complexity

1.1.1 Cost of a HCOD

We first consider a matrix $A$ of size $m \times n$. The LQ decomposition of $A = LQ$ requires $\sum_{i=1}^{\min(m,n)} (n-i) \approx nm$ Givens rotations Golub & Van Loan (1996a), each of them being applied on $A$ and $Q$. The cost to obtain $L$ is then $nm^2$ elementary operations, while the cost for $Q$ is $mn^2$.

A COD $[V \ U] \begin{bmatrix} 0 & 0 \\ L & 0 \end{bmatrix} [Y \ Z]^T$ starts with a LQ before applying a QL on the obtained $AY$. The cost of the QL is $(m-r)r^2$ to obtain $L$ and $(m-r)m^2$ to obtain $W = [V \ U]$. The total cost of the COD is roughly $n^2m + nm^2 + (m-r)m^2$. When $m$ is much smaller than $n$ (typically for $A_1$), the cost is in $O(mn^2)$ while when $m$ and $n$ have the same magnitude order (typically for $A_p$), the cost is in $O(2n^3)$.

We now consider the set of $p$ matrices $A_k$, each of them being of size $m_k \times n$, and of rank in the hierarchy $r_k = r_{k-1} - 1$. The size of $A_p$ is $m = \sum m_k$. For simplicity, we consider that all the decompositions are computed using a series of Givens rotations, and that all the basis are computed explicitly into a dense format.

The HCOD can be computed by a set of COD. For each COD, the coefficients of the intermediary bases $Z_k$ are not necessary: only the $A_{k+1}Z_k$ are needed. The $Y_k$ and $Z_k$ are kept as product of elementary transformations. The number of operations to get the right part of the decomposition $Y$ is the same as for getting the right part of decomposition of the COD. The QL decompositions $W_k$ are then made for each level $k$ independently, and the cost is each time the same as for the QL of the COD of $A_k$. The total cost is then:

$$c_{\text{HCOD}} = n^2m + nm^2 + \sum_{k=1}^{p} (m_k - r_k)m_k^2 \quad (1)$$

The cost is less than the COD of $A_p$, but more than its QR. Of course, if all the matrices are full-row rank, the three costs are the same.

1.1.2 Cost of the eHQP

We consider now the two primal and dual algorithms $\pi-1$ and $\pi-2$. Since the slack variable $w_p^*$ can be computed in both, we remove the redundant computations from Alg. $\pi-1$. 

2
The computation involved by the primal optimum amounts to the forward substitution of the stacked $U_k b_k$ by the triangular matrix formed of the $M_k$ and $L_k$. The cost is $\sum m_k r_k$ for computing the $U_k b_k$ and then $r_p^2$ for the forward substitution. The total cost is roughly $r_p^2$.

The computation of the multipliers of level $k$ is similar: it is the backward substitution of the $\rho$, whose own cost is negligible with respect to the cost of the substitution. The cost is then $r_k^2$. The cost to compute $\Lambda_p$ is then $\sum r_k^2 \approx r_p^3$ which is similar to the cost of the whole HCOD.

In conclusion, if the proposed method is used to compute only the primal optimum of an eHQP, the cost is similar to the cost of the inversion of a full-rank system of the same size using a QR decomposition and a forward substitution.

If both primal and dual optima are needed (for example in the active search), the computation of the multipliers $\Lambda_p$ is costly and should be parsimoniously realized.

1.1.3 Updating the decomposition

The major cost in the eHQP is the decomposition. When changing one element of the active set, it is possible to only update the HCOD without recomputing all the decomposition from scratch. The choice of a triangular decomposition is very suitable for such an update, as shown in Golub & Van Loan (1996b). The same update can be adapted for the hierarchical decomposition. Similarly to the HCOD cost that is equivalent to the COD cost (roughly $2n^3$), the same reasoning can apply to the HCOD updates, that are equivalent in cost to the COD update. They are similar to one of the $m$ steps of the decomposition: the total cost is then in $O(n^2 + mn)$.

1.1.4 Step length

Finally, denoting by $s_k \geq m_k$ the total number of constraints of level $k$ (i.e. active and inactive), the step length computation for each level requires the multiplication of the rows of $A_k$ for the $s_k - m_k$ inactive constraints. The cost is $\sum_k n(s_k - m_k) = n(s - m)$ with $s = \sum_k s_k$.

1.1.5 Cost of the HQP active search

The cost of the active search depends of course on the number of iterations in the inside loop (the number of iterations of the outside loop being bounded). If during an iteration a constraint is activated, the iteration implies the computation of the primal, of the step and an update. If the iteration is
Table 1: Computation cost for each operation of the active search. The approximations are given for $n = m$ and $k = p$

<table>
<thead>
<tr>
<th>operation</th>
<th>cost</th>
<th>approx.</th>
</tr>
</thead>
<tbody>
<tr>
<td>QR</td>
<td>$nm^2 + n^2m$</td>
<td>$2n^3$</td>
</tr>
<tr>
<td>COD</td>
<td>$n^2m + nm^2 + (m - r)m^2$</td>
<td>$2n^3$</td>
</tr>
<tr>
<td>HCOD</td>
<td>$n^2m + nm^2 + \sum(m_k - r_k)m_k^2$</td>
<td>$2n^3$</td>
</tr>
<tr>
<td>Primal $x_p^*$</td>
<td>$\frac{r^2}{p}$</td>
<td>$n^2$</td>
</tr>
<tr>
<td>Dual $w_k^*, \Lambda_k$</td>
<td>$\frac{r_k^2}{p}$</td>
<td>$n^2$</td>
</tr>
<tr>
<td>Dual $\Lambda_p$</td>
<td>$\sum_{k=1}^{p} \frac{r_k^2}{p}$</td>
<td>$pn^2$</td>
</tr>
<tr>
<td>Update</td>
<td>$n^2 + mn$</td>
<td>$2n^2$</td>
</tr>
<tr>
<td>Step $\tau$</td>
<td>$n(s - m)$</td>
<td>$n(s - n)$</td>
</tr>
</tbody>
</table>

concluded with a deactivation, it implies the primal, the step, the dual and the update. Finally, due to the outer loop of Alg. π-3 there are $p$ iterations that do not activate nor deactivate any constraints. We denote by $N_U$ and $N_D$ the number of activations and deactivations of the algorithm. The total cost is easily deduced from the above list and Table 1.

If we suppose additionally that the hierarchical problem constraints the whole parameter space ($m = n$) and if we smooth the differences between the level for the dual computation and the updates (approximate costs are given on the third column of Table 1), the total cost is:

$$c_{asearch} = 2n^3 + (3N_U + 4N_D + 2p)n^2 + (N_U + N_D + p)n(s - n)$$

In particular, the minimum cost of the active search is obtained for $N_U = N_D = 0$:

$$c_{asearch} = 2n^3 + np(n + s)$$

In that last cost, compared to a standard QP, the hierarchy brings the $p$ factor in the last term. This is mainly due to the bi-dimensional structure of the multipliers $\Lambda_p$.

1.2 Conclusion

Without surprise, the cost of the active-search strongly depends on the number of activations and deactivations $N_U + N_D$ that are needed to reach the optimal active set. If a good approximation of the optimal active set
is known, then the total cost of the active search can be significantly de-
creased. In particular, if a similar problem has already been solved, then
the obtained optimal active set can be reused to warm-start the current
problem. When using a bound on the number of iterations of the active-set
loop \((N_U + N_D << m ≤ n)\), the total resolution cost is dominated by the
initial HCOD computation and is equivalent to \(2n^3\).

2 Continuity

In this section, we recall the continuity property and give the proof. We
consider a continuous parametric optimization problem, depending on the
parametric variable \(t\):

\[
\text{lex min}_{x, w_1 \ldots w_p} \{ \|w_1\|, \|w_2\|, \ldots, \|w_p\| \}.
\]

subject to \(\forall k = 1 : p, A_k(t)x \leq b_k(t) + w_j\)

with \(A_k(t)\) and \(b_k(t)\) continuous function of a real parameter \(t \in \mathbb{R}\). This
problem is denoted by HQP(\(t\)). We denote by \(S^*_0\) the optimal active set.

**Theorem π-4.1** At a given \(t_0\), if \(t \rightarrow x^*_{S^*_0}(t)\) is continuous, then \(x^*(t)\) is
continuous in \(t_0\).

The theorem is proved by first proving the continuity of the eHQP opti-
mum.

2.1 Equality-only problem

First, the eHQP optimum is known to be continuous outside of the singular
regions. This is formalized by the following result. The functions giving the
rank of the projected matrices are denoted by:

\[
r_k : t \rightarrow r_k(t) = \text{rank}(A_k(t)) - \text{rank}(A_{k-1}(t))
\]

**Lemma 2.1.** Consider a eHQP of the form (2). For a given \(t_0\), if the
maps \(r_1 \ldots r_p\) are constant on a neighborhood of \(t_0\), then \(O(t)\) and \(x^*(t)\) are
continuous at \(t_0\).

**Proof.** The result is straightforward using the fact that the pseudo-inverse
map \(A \rightarrow A^+\) is continuous inside the set of constant rank of \(A\) Ben-Israel
& Greville (2003).
When the $r_k(t)$ are not constant, the solution maps are not continuous in general. The only case where the continuity is not ensured is when passing inside a singular region and only at the region border. Inside the singularity, the optimum remains continuous.

2.2 Active-set continuity

Consider now the full problem (2) with inequalities. It is possible to show that, apart from these discontinuities due to the singularity of the eHQP, the continuity of the optimum is ensured. $S^*(t)$ denotes the optimal active set of HQP$(t)$ corresponding to $x^*(t)$. For a given active set $S$, the eHQP associated to (2) is denoted by eHQP$(t, S)$ and the optimum of this eHQP is denoted by $x^*_S(t)$. We can now prove Theorem π-4.1.

Proof. If the optimal active set is constant in a neighborhood of $t_0$, the result is straightforward using Th. 2.1. We then consider the case where the active set changes on $t_0$: $\forall t > t_0, S(t) \neq S(t_0)$. We suppose that the active set of level $k$ is increased by one constraint $A_{up}x \leq b_{up}$ in $t > t_0$ (the case with multiple activations or deactivations follows naturally). Since $(A_{up}, b_{up})$ is active for any $t > t_0$ of a neighborhood, it means the optimum lies on this constraint at $t_0$: $A_{up}x^*_p(t_0) = b_{up}$. Both active sets $S(t_0)$ and $S_+ = S(t_0) \cup \{(A_{up}, b_{up})\}$ give the same $x^*_p$. Using the continuity of the eHQP associated to $S_+$ the continuity of $x^*(t)$ toward $x^*(t_0)$ is straightforward. □

2.3 Remarks

The continuity of the solution before and after the active-set changes can be understood by looking at the structure of the HCOD and the changes applied to it when inserting a new row. Consider the row to be added at level $k$ of the decomposition. The rank of level $k$ is denoted by $r_k$; the total rank of the HCOD is denoted by $r_p = \sum_{k=1}^{p} r_k$. Three cases can happen, depending on the row to be added to the decomposition:

- $r_k$ remains constant (in which case, $r_p$ remains constant);
- $r_k$ increases and $r_p$ increases too;
- $r_k$ increase and $r_p$ remains constant (in which case, the rank $r_j$ of one level $j > k$ decreases).

The continuity of the optimum appears in the components of $y^*$. In the first case, only the continuity of $y^*_k$ is questionable. It is ensured by
Fig. 1: Continuity and stability: when the system approaches a boundary, the optimum continuously changes to a pure motion along the boundary, until it finally stably stops on the limit. The optimum does not fight against the reference required by the equality constraint of level 2.

the continuity of the partitioned generalized inverse Ben-Israel & Greville (2003):

\[
\begin{bmatrix}
A \\
C
\end{bmatrix}
\begin{bmatrix}
b \\
d
\end{bmatrix} \rightarrow \begin{bmatrix}
d \rightarrow CA^+b
\end{bmatrix}
\]

for any \(A, b, C\). The continuity of the \(y^*_j, j > k\) follows, using recursively the continuity of \(y^*_{j-1}\) and the continuity of the eHQP.

If both \(r_k\) and \(r_p\) increase, the activation adds a new column \(Y_{up}\) in \(Y_k\). This vector belongs to the null space of the HCOD at \(t_0\): \(Y_{up}^T Y_p(t_0) = 0\). This case is then equivalent to considering that \((A_{up}, b_{up})\) was added as a new least-priority level \(p + 1\) of the hierarchy. The \(y^*_p\) is continuous using Th. 2.1. And \(y^*_{p+1}\) is continuously evolving from 0 using (4).

Finally, the key case is when the rank increase at level \(k\) and causes a rank loss at level \(j > k\). It means that the new row is linked with level \(j\). In that case, the corresponding DOF was allocated to the level \(j\) before \(t_0\). At \(t_0\), the constraint denoted by \(A_{up}x \leq b_{up}\) is reached and activated. The DOF is then reallocated to the level \(k\). The reallocation is performed when the optimum is exactly on the constraint, thus causing no discontinuity. This case is depicted in Fig. 1.

3 Stability

We consider the IK control law given in (π-70) and recalled here:
\[ \dot{e}_k = J_k \dot{q} \]  

(5)

\[ \forall k \in S_I, \quad \dot{e}_k \leq b_k \]  

(6)

\[ \forall k \in S_E, \quad \dot{e}_k = \dot{e}_k^* \]  

(7)

where \( S_I \cup S_E \) is a partition of the set \( \{1 \ldots p\} \) of the \( p \) first integers: \( S_I \) are the task levels that are defined by a limit \( b_k \) and \( S_E \) are the task constraints to follow a given velocity \( \dot{e}_k^* \). When the task function \( e_k \) is an error between a reference and a current sensor value, the task reference \( \dot{e}_k^* \) is often given to drive the error to 0:

\[ \dot{e}_k^* = -\kappa_k e_k \]  

(8)

where \( \kappa_k > 0 \) is a gain used to tune the convergence speed. Finally, we suppose that 0 is an acceptable solution for all the tasks of \( S_I \), i.e. \( \forall k \in S_I, \quad b_k > 0 \). The robot input \( \dot{q} \) is then computed as the result of a HQP:

\[
\text{lex min}_{\dot{q}, w_1 \ldots w_p} \{ \|w_1\|, \|w_2\|, \ldots, \|w_p\| \}.
\]  

(9)

subject to (5), (6), (7). We prove in the following the stability of this control scheme. The following notation is used:

\[ e_E = \begin{bmatrix} e_{k1} \\ \vdots \\ e_{k_E} \end{bmatrix} \]  

(10)

for \( S_E = \{k_1 \ldots k_E\} \), and by \( J_E \) the associated jacobian.

**Theorem π-5.1** The hierarchical IK control law is stable in the sense of Lyapunov. It is asymptotically stable iff \( J_E \) is full row rank and none of the equality-constraint levels of the HCOD are rank deficient.

**Proof.** We define the Lyapunov energy function by \( V = \frac{1}{2} e_E^T e_E \). Then

\[ \dot{V} = e_E^T \dot{e}_E = e_E^T J_E \dot{q} \]  

(11)

where \( \dot{q} \) is computed from the eHQP of the current optimal active search:

\[ \dot{q} = Y H_p^T W_p^T b_p \]  

(12)
Suppose that the active set of $S$ is built incrementally from the initial guess $S_E$. The initial decomposition involves only $J_E$, with the following notations:

$$J_E = W_E H_E Y_E^T$$

(13)

For each active component of the inequality levels $S_I$, there are two options: it corresponds to a column $Y_i$ of $Y_E$ (the constraint is linearly linked to $E$) or it is decoupled. In the second case, the component can be equivalently activated at the least-priority level. The optimum is then equivalently written:

$$y^* = (y_1^*, \ldots, y_p^*, y_{p+1}^*)$$

(14)

with $y_{p+1}^*$ the equivalent level where all the decoupled constraints of $S$ are set and all the active inequalities of levels 1 to $p$ are coupled with $J_E$: both bases $Y$ and $Y_E$ are the same with a permutation $\Pi$ on the column order: $Y = Y_E \Pi$. The optimum $y_p^*$ of the full iHQP can be computed from the optimum of the eHQP corresponding to $J_E$, denoted by $y_E^*$:

$$y_p^* = \left[ \begin{array}{c} y_1^* \\ \vdots \\ y_p^* \end{array} \right] = \Pi^T \left[ \begin{array}{c} (1 - \tau_1)y_{E1}^* \\ \vdots \\ (1 - \tau_p)y_{Ep}^* \end{array} \right] = \Pi^T \Delta_\tau y_E^*$$

(15)

where the $\tau_k$ are the step lengths met during the update phases and $\Delta_\tau$ is the diagonal matrix of the $1 - \tau_k$.

Introducing this last form into (11):

$$\dot{V} = e_E^T W_E H_E \Delta_\tau y_E^*$$

$$= e_E^T W_E H_E \Delta_\tau H_E^+ W_E^T e_E$$

(16)

(17)

which is non-negative since $\Delta_\tau$ is positive and $H_E^+$ respects the three first conditions of Moore-Penrose.

If $J_E$ is full rank, all the $N$ of $H_k$ are empty. Moreover, if none of the equality levels of the iHQP are rank deficient, then it is as if all the inequalities had been activated at a least-priority level $p$. Then:

$$\dot{V} = e_E^T e_E > 0$$

(18)

which ensures the asymptotic stability.
Acknowledgments

This work was supported by grants from the RobotHow.cog EU CEC project, Contract No. 288533 under the 7th Research program (www.robohow.eu), and FUI Project ROMEO.

References


