Supplement: Efficient Decomposed Learning for Structured Prediction

Rajhans Samdani RSAMDAN2@ILLINOIS.EDU
Dan Roth DANR@ILLINOIS.EDU

This supplement provides the proof for all the theorems and corollaries in the main paper. We provide the proofs as well as the theorem statements for the ease of understanding while borrowing the notation from the main draft.

Theorem 1. *DecL is exact if* $\forall \mathbf{w} \in W^*, \exists \epsilon > 0$, *such that* $\forall \mathbf{w}' \in B(\mathbf{w}, \epsilon)$, $\forall (\mathbf{x^j}, \mathbf{y^j}) \in D$ *the following condition holds for* $nbr(\mathbf{y^j})$*: if* $\exists \mathbf{y} \in \mathcal{Y}$ *with* $f(\mathbf{x^j}, \mathbf{y}; \mathbf{w}') + \Delta(\mathbf{y^j}, \mathbf{y}) > f(\mathbf{x^j}, \mathbf{y^j}; \mathbf{w}')$ *then* $\exists \mathbf{y}' \in nbr(\mathbf{y^j})$ *with* $f(\mathbf{x^j}, \mathbf{y}'; \mathbf{w}') + \Delta(\mathbf{y^j}, \mathbf{y}') > f(\mathbf{x^j}, \mathbf{y^j}; \mathbf{w}')$.

Proof. Assume that we are given an $\epsilon > 0$ for which the condition mentioned in the theorem statement holds true (for the sake of simplicity we assume the same ϵ for all $w \in \mathbb{R}^d$ by taking a minimum over all possible ϵ values.) Since $W^* \subseteq W^{dec}$ (Observation 3 in the paper), to show exactness, it is sufficient to show $W^{dec} \subseteq W^*$. We want to show that $\forall \overline{\mathbf{w}} \in \mathbb{R}^d$, if $\overline{\mathbf{w}} \in W^{dec}$ then $\overline{\mathbf{w}} \in W^*$. Suppose there exists a $\overline{\mathbf{w}} \in W^{dec}$ with $\overline{\mathbf{w}} \notin W^*$; we will show by contradiction that no such $\overline{\mathbf{w}}$ exists.

Consider any $\mathbf{w}^* \in W^*$ (recall that we assume that W^* is non-empty) and define $\mathbf{w_t} = \mathbf{w}^* + t\overline{\mathbf{w}}$ for $t \in [0,1]$ By convexity of W^{dec} (Observation 2), $\mathbf{w_t} \in W^{dec}$ $\forall t \in [0,1]$. Define $m = \max\{t \in [0,1] | \mathbf{w_t} \in W^*\}$. By closedness of W^* (Observation 1) and by our assumption that $\overline{\mathbf{w}} \notin W^*$, we get $m < 1$. Now we have that for all $\epsilon' \in (m, 1]$, $\mathbf{w_{m+\epsilon'}} \notin W^*$ and $\mathbf{w_{m+\epsilon'}} \in W^{dec}$. It is easy to verify that for a given ϵ, $\|\mathbf{w_{m+\epsilon'}} - \mathbf{w_m}\| \leq \epsilon$ for an appropriate value of ϵ' ($\epsilon' \leq \frac{\epsilon}{\|\overline{\mathbf{w}}\|}$). Let the corresponding weight vector be $\mathbf{w}' = \mathbf{w_{m+\epsilon'}} \in W^{dec}$. Note that since $\|\mathbf{w}' - \mathbf{w_m}\| \leq \epsilon$, we get by the condition mentioned in the theorem statement that if $\exists \mathbf{y}$ with $f(\mathbf{x^j}, \mathbf{y}; \mathbf{w}') + \Delta(\mathbf{y^j}, \mathbf{y}) > f(\mathbf{x^j}, \mathbf{y^j}; \mathbf{w}')$ then $\exists \mathbf{y}' \in nbr(\mathbf{y^j})$ with $f(\mathbf{x^j}, \mathbf{y}'; \mathbf{w}') + \Delta(\mathbf{y^j}, \mathbf{y}') > f(\mathbf{x^j}, \mathbf{y^j}; \mathbf{w}')$.

In order to prove by contradiction that our assumption is wrong and that no such $\overline{\mathbf{w}}$ exists, it is sufficient to show that $\mathbf{w}' \notin W^{dec}$. This easily follows as

$$\begin{aligned}
&\mathbf{w}' \notin W^* \Rightarrow l(\mathbf{w}') > 0 \text{ (due to closedness of } W^*)\\
&\Rightarrow \exists (\mathbf{x^j}, \mathbf{y^j}) \in D, f(\mathbf{x^j}, \mathbf{y_{w'}}(\mathbf{x^j}); \mathbf{w}'') + \Delta(\mathbf{y^j}, \mathbf{y_{w'}}(\mathbf{x^j})) \\
&\qquad > f(\mathbf{x^j}, \mathbf{y^j}; \mathbf{w}')\\
&\Rightarrow \exists (\mathbf{x^j}, \mathbf{y^j}) \in D, \exists \mathbf{y}' \in nbr(\mathbf{y^j}),\ f(\mathbf{x^j}, \mathbf{y}'; \mathbf{w}') + \Delta(\mathbf{y^j}, \mathbf{y}') \\
&\qquad > f(\mathbf{x^j}, \mathbf{y^j}; \mathbf{w}')\\
&\Rightarrow DecL(\mathbf{w}'; D) > 0 \Rightarrow \mathbf{w}' \notin W^{dec}\ ,
\end{aligned}$$

which completes the proof. □

Corollary 1. *DecL is exact if* Δ *is subadditive and* $\forall \mathbf{w} \in W^*, \exists \epsilon > 0$ *such that* $\forall \mathbf{w}' \in B(\mathbf{w}, \epsilon)$, $\forall (\mathbf{x^j}, \mathbf{y^j}) \in D$, $\forall \mathbf{y} \in \mathcal{Y}$, $s(\mathbf{y}, \mathbf{y^j})$ *can be partitioned into sets* $s_1, \ldots, s_l$ *such that* $\forall k \in \{1, \ldots, l\}$, $(\mathbf{y}_{s_k}, \mathbf{y^j}_{-s_k}) \in nbr(\mathbf{y^j}, \mathcal{S}^j)$ *and*

$$f(\mathbf{x^j}, \mathbf{y}; \mathbf{w}') - f(\mathbf{x^j}, \mathbf{y^j}; \mathbf{w}') \leq \sum_{k=1}^{l} \left(f(\mathbf{x^j}, (\mathbf{y}_{s_k}, \mathbf{y^j}_{-s_k})); \mathbf{w}') - f(\mathbf{x^j}, \mathbf{y^j}; \mathbf{w}') \right) \quad .(1)$$

Proof. Consider any $\mathbf{y^j}$ and $\mathbf{y} \in \mathcal{Y}$. Assume for a given w', $s(\mathbf{y^j}, \mathbf{y})$ can be partitioned into sets $s_1, \ldots, s_l$ such that $(\mathbf{y}_{s_k}, \mathbf{y^j}_{-s_k}) \in nbr(\mathbf{y^j}, \mathcal{S}^j)$ and such that condition (1) holds. Now we have

$$\begin{aligned}
&\sum_{k=1}^{l} \left(f(\mathbf{x^j}, (\mathbf{y}_{s_k}, \mathbf{y^j}_{-s_k})) - f(\mathbf{x^j}, \mathbf{y^j}) + \Delta(\mathbf{y^j}, (\mathbf{y}_{s_k}, \mathbf{y^j}_{-s_k})) \right)\\
\geq\ & f(\mathbf{x^j}, \mathbf{y}; \mathbf{w}') - f(\mathbf{x^j}, \mathbf{y^j}; \mathbf{w}') + \sum_{k=1}^{l} \Delta(\mathbf{y^j}, (\mathbf{y}_{s_k}, \mathbf{y^j}_{-s_k}))\text{(from (1))}\\
\geq\ & f(\mathbf{x^j}, \mathbf{y}; \mathbf{w}') - f(\mathbf{x^j}, \mathbf{y^j}; \mathbf{w}') + \Delta(\mathbf{y^j}, \mathbf{y}) \ (\Delta \text{ subadditive})
\end{aligned}$$

Thus if $f(\mathbf{x^j}, \mathbf{y}) - f(\mathbf{x^j}, \mathbf{y^j}) + \Delta(\mathbf{y^j}, \mathbf{y}) > 0$ then for some s_k, $f(\mathbf{x^j}, (\mathbf{y}_{s_k}, \mathbf{y^j}_{-s_k})) - f(\mathbf{x^j}, \mathbf{y^j}) + \Delta(\mathbf{y^j}, (\mathbf{y}_{s_k}, \mathbf{y^j}_{-s_k})) > 0$ which is the required condition in Theorem 1. Since $(\mathbf{y}_{s_k}, \mathbf{y^j}_{-s_k}) \in nbr(\mathbf{y^j})$, this completes the proof. □

Corollary 2. *If* $\mathcal{Y}$ *is specified by* k `OR` *constraints, then Decl-*$(k+1)$ *is exact for subadditive* Δ.

Proof. Let $\mathcal{Y}$ be specified by k `OR` constraints: $C_1(\mathbf{y}) = 1, \ldots, C_k(\mathbf{y}) = 1 \Leftrightarrow \mathbf{y} \in \mathbf{Y}$. We will show that DecL-$(k+1)$ satisfies Cor. 1 and hence is exact. We will provide a constructive proof where we will generate appropriate decompositions.

Consider any two $\mathbf{y^1}, \mathbf{y^2} \in \mathcal{Y}$. We will generate sets $s_1, \ldots, s_l$ which a) partition $s(\mathbf{y^1}, \mathbf{y^2})$ with $|s_i| \leq k+1$ $\forall i = 1, \ldots, l$, b) satisfy $(\mathbf{y^2}_{s_i}, \mathbf{y^1}_{-s_i}) \in \mathcal{Y}$ $\forall i =$

$1,\ldots,l$ (i.e. is feasible), and c) satisfy

$$\begin{aligned} & f(\mathbf{x},\mathbf{y^2};\mathbf{w}) - f(\mathbf{x},\mathbf{y^1};\mathbf{w}) \\ = & \sum_{i=1}^{l} f(\mathbf{x},(\mathbf{y^2}_{s_i},\mathbf{y^1}_{-s_i});\mathbf{w}) - f(\mathbf{x},\mathbf{y^1};\mathbf{w}) \ . \end{aligned}$$

Let s^1 the set of all indices in $s(\mathbf{y^1},\mathbf{y^2})$ such that flipping the bit in $\mathbf{y^1}$ corresponding to any index in s^1 will violate some constraints: $s^1 = \{i \in s(\mathbf{y^1},\mathbf{y^2}) | (y^2_i, \mathbf{y^1_{-i}}) \notin \mathcal{Y}\}$. Let $s^2 = s(\mathbf{y^1},\mathbf{y^2}) \setminus s^1$ be the rest of the bits. Also for each $i \in s^1$, consider the set of constraints violated by flipping the bit y^1_i: $C(i) = \{C_t | 1 \le t \le k, C_t\left((y^2_i, \mathbf{y^1_{-i}})\right) \neq 1\}$.

Now, we present an algorithmic recipe for generating the required sets $s_1,\ldots,s_l$ which proceeds by flipping bits in $\mathbf{y^1}$ given by $s(\mathbf{y^1},\mathbf{y^2})$.

1. Initialize $i = 0$.
2. If $s^1 = \emptyset$, go to step 5. Else, do $i = i+1$, pick some $t \in s^1$ and remove t from s^1; let $s_i = \{t\}$; flip the bit y^1_t in $\mathbf{y^1}$, resulting in some constraints getting violated.
3. In order to satisfy any violated constraints, flip a **minimal** set of bits from $s(\mathbf{y^1},\mathbf{y^2}) \setminus s_i$ such that all the constraints are satisfied. Add the set of bits which are flipped to s_i.
4. If none of the sets in $s_1,\ldots,s_{i-1}$ overlap with s_i, then go to step 2. Otherwise, merge all sets in $s_1,\ldots,s_{i-1}$ overlapping with s_i into s_i; set i to be the total number of existing sets remained after merging (including the current set under consideration) and relabel the existing sets to $s_1,\ldots,s_i$, with s_i be the current set in consideration. Flip all the bits in $\mathbf{y^1}$ given by s_i and repeat step 3 to satisfy any violated constraints.
5. Suppose the current sets obtained after all variables in s^1 are considered are $s_1,\ldots,s_i$ (for the vacuous case of $i=0$, no sets are considered.) Let $s^3 = s^2 \setminus (\cup_{u=1}^{i} s_u)$ which are the rest of the variables which haven't yet been considered and which don't violate any constraint when flipped. We thus distribute each of these variables into singleton sets — $s_{i+1},\ldots,s_l$ where $l = i + |s^3|$.
6. Return $s_1,\ldots,s_i,s_{i+1},\ldots,s_l$

Note that, we have to merge the sets sharing a variable in step 4 as we are trying to creating a partition of $s(\mathbf{y^1},\mathbf{y^2})$. The main task is to show that sizes of the sets don't exceed $k+1$ after merging. To show that the above procedure generates the required decompositions, we make a sequence of easy observations.

1. Steps 1-4 are guaranteed to converge since $\mathbf{y^2} \in \mathcal{Y}$ (so in worst case, we can include all the bits in $s(\mathbf{y^1},\mathbf{y^2})$ into one set.)
2. To satisfy a violated constraint, at most one bit needs to be flipped (as the constraints are `OR`).
3. If a flipped bit contributes a '1' to a constraint thereby satisfying it, the constraint remains satisfied regardless of any other bits.
4. Take the above two observations and consider step 3 and 4 of the algorithm. If by flipping bits in s_i, a constraint C_t is satisfied, then C_t remains satisfied when bits in the set $s_i \cup s_j$ are flipped. In other words, when looping between steps 3 and 4, any constraint in the set of all constraints, $\{C_1, C_2, \ldots, C_k\}$, is violated at most once.
5. The last observation, combined with observation 2, implies that at most one variable per constraint is flipped during steps 3 and 4. Step 3 flips a minimal set of bits to satisfy the constraints, which coupled with above observation implies that s_i, at any stage in the loop, contains at most $1+k$ variables — one corresponding to step 2 plus at most k corresponding to each constraint.
6. $(\mathbf{y^2}_{s_u},\mathbf{y^1}_{-s_u}) \in \mathcal{Y}$ for $1 \le u \le i$ by construction (step 3 ensures that all the constraints are satisfied) and $(\mathbf{y^2}_{s_u},\mathbf{y^1}_{-s_u}) \in \mathcal{Y}$ for $i+1 \le u \le l$ by definition of s^3.

Thus we have found the required partition of $s(\mathbf{y^1},\mathbf{y^2})$ into $s_1,\ldots,s_l$ with $|s_i| \le k+1, \forall i$. □

Corollary 3. *If $\mathcal{Y}$ is specified by k linear constraints: $A\mathbf{y} \le b$ (or '$\ge$', '$=$'), where A is a binary matrix such that any two variables in $\mathbf{y}$ participate in at most one constraint, Decl-$3k$ is exact for subadditive Δ.*

Proof. The proof is similar to the proof of Cor. 2. We provide the proof for $A\mathbf{y} \le b$; the cases for $\ge$ and $=$ can be dealt with similarly. We will show that DecL-$3k$ satisfies Cor. 1 and hence is exact.

To show this, consider any two $\mathbf{y^1},\mathbf{y^2} \in \mathcal{Y}$. We will generate sets $s_1,\ldots,s_l$ which a) partition $s(\mathbf{y^1},\mathbf{y^2})$ with $|s_i| \le 3k\ \forall i = 1,\ldots,l$, b) satisfy $(\mathbf{y^2}_{s_i},\mathbf{y^1}_{-s_i}) \in \mathcal{Y},\ \forall i = 1,\ldots,l$ (i.e. is feasible), and c) satisfy

$$\begin{aligned} & f(\mathbf{x},\mathbf{y^2};\mathbf{w}) - f(\mathbf{x},\mathbf{y^1};\mathbf{w}) \\ = & \sum_{i=1}^{l} f(\mathbf{x},(\mathbf{y^2}_{s_i},\mathbf{y^1}_{-s_i});\mathbf{w}) - f(\mathbf{x},\mathbf{y^1};\mathbf{w}) \ . \end{aligned}$$

First we partition the bit-indices in $s(\mathbf{y^1},\mathbf{y^2})$ as: $s^0 = \{i \in s(\mathbf{y^1},\mathbf{y^2}) | y^1_i = 0\}$ and $s^1 = \{i \in s(\mathbf{y^1},\mathbf{y^2}) | y^1_i =$

1}. That is s^0 contains all the bits which have been flipped from 0 to 1 from $\mathbf{y^1}$ to $\mathbf{y^2}$ and s^1 contains the bits which have been flipped from 1 to 0. Now, we present a recipe for generating the required sets $s_1, \ldots, s_l$ which proceeds by flipping bits in $\mathbf{y^1}$ given by $s(\mathbf{y^1}, \mathbf{y^2})$.

1. Initialize $i = 0$, and copy $s_c^1 = s^1$. Throughout the algorithm, s_c^1 would hold those bits in s^1 which haven't been considered even once.

2. If $s^0 = \emptyset$, go to step 7. Otherwise, do $i = i + 1$, pick any element in $t \in s^0$, and remove it from s^0. Let $s_i = \{t\}$. Flip y_t^1 in $\mathbf{y^1}$ and if $(y_t^2, \mathbf{y^1}_{-t}) \in \mathcal{Y}$, then repeat this step as we don't need to flip more bits.

3. In order to satisfy any constraints violated by flipping bit(s) in $\mathbf{y^1}$, check if it is possible to flip a **minimal** set of bits from s_c^1 such that all the constraints are satisfied. If this is true then add all the flipped bits to s_i and do $s_c^1 = s_c^1 \setminus s_i$, and go to step 2. If this is not possible; i.e. in order to satisfy all constraints it is needed to flip bits from outside set s_c^1, then go to step 4.

4. Since it is not possible to satisfy all the constraints violated after flipping y_t by flipping bits from s_c^1, flip a **minimal** set of bits from s^1 to satisfy the constraints. Add these bits to s_i. Do $s_c^1 = s_c^1 \setminus s_i$.

5. Merge all sets in $s_1, \ldots, s_{i-1}$ overlapping with s_i into s_i; set i to be the total number of existing sets remained after merging (including the current set under consideration) and relabel the existing sets to $s_1, \ldots, s_i$, with s_i being the current set in consideration.

6. Now flip all the bits in $\mathbf{y^1}$ corresponding to s_i. If this does not violate any constraints, go to step 2, else go to step 3 to satisfy any violated constraints.

7. Suppose the final resulting sets, after we have considered all variables in s^0, are $s_1, \ldots, s_i$. Let $s^3 = s^1 \setminus (\cup_{u=1}^{i} s_u)$ which are the rest of the variables which haven't yet been considered. Since flipping variables in s^1 cannot violate any constraint (all coefficients of A are positive), we distribute each of these variables into singleton sets — $s_{i+1}, \ldots, s_l$ where $l = i + |s^3|$.

8. Return $s_1, \ldots, s_i, s_{i+1}, \ldots, s_l$

Note that, we have to merge the sets sharing a variable in step 4 as we are trying to creating a partition of $s(\mathbf{y^1}, \mathbf{y^2})$. The main task is to show that sizes of the sets don't exceed $3k$ after merging. To show that the above procedure generates the required decompositions, we make a sequence of easy observations.

1. Steps 1-6 are guaranteed to converge since $\mathbf{y^2} \in \mathcal{Y}$ (so in the worst case, we can include all the bits in $s(\mathbf{y^1}, \mathbf{y^2})$ into one set.)

2. Flipping any bit in s^1 cannot result in a constraint violation (flipping a bit from 1 to 0 cannot violate a constraint as all coefficients in A are positive.)

3. Since A is a binary matrix, so if flipping a variable in s^0 causes a constraint violation (in step 3 and 4), it can violate the constraint by at most 1 and thus at most one variable in s^1 needs to be flipped to satisfy the constraint again.

4. Now lets analyze the case in step 3 when it is not possible to satisfy constraints by flipping bits in s_c^1. It must be the case that in order to satisfy the violated constraints, we need to flip a variable which is also needed to satisfy the constraint violated when some other $t' \in s^0$ was flipped i.e. they share a common variable needed to satisfy some constraints. Such a "clash" over flipping one variable cannot occur to satisfy a constraint where both y_t and $y_{t'}$ both participate because in this case, at least two, and not one, variable from s^1 must participate in this constraint which can then both be flipped to satisfy this constraint.

   Thus, by this observation, in step 4, we have a one-to-many mapping for each $t \in s_i \cap s^0$ to the set of constraints such that each variable $t \in s_i$ maps to the set of constraints which could not be satisfied in step 3. Thus if k' be the maximum number of variables in any $s_i \cap s^0$ for any i, then $k' \leq k$ — at most one per constraint.

5. If we look at the set of all the constraints which contain only one variable from $s_i \cap s^0$, then for these constraints, at most one element from s^1 needs to be flipped per constraint due to the third observation above. If we consider those constraints which contain more than one variable from $s_i \cap s^0$, then due to the given restriction that any two variables can occur together in at most one constraint, a total of at most k' additional variables in s^1 need to be flipped in order to satisfy these constraints.

6. Consequently, since steps 3 and 4 flip a minimal number of bits to satisfy the constraints, we get that $|s_i| \leq 2k' + k \leq 3k$.

7. $(\mathbf{y^2}_{s_u}, \mathbf{y^1}_{-s_u}) \in \mathcal{Y}$ for $1 \leq u \leq i$ by construction (step 3 ensures that all the constraints are satisfied) and $(\mathbf{y^2}_{s_u}, \mathbf{y^1}_{-s_u}) \in \mathcal{Y}$ for $i+1 \leq u \leq l$ by definition of s^3.

Thus we have found the required partition of $s(\mathbf{y^1}, \mathbf{y^2})$ into $s_1, \ldots, s_l$ with $|s_i| \leq 3k, \forall i$. □

Theorem 2. *For PMNs where Assumption 2 is satisfied and with subadditive Δ, DecL with $\mathcal{S}_{pair}$ is exact.*

Proof. We show that $\mathcal{S}_{pair}$ satisfies Cor. 1 and hence leads to exactness. Consider some $\mathbf{w}^* \in W^*$, and $\mathbf{w} \in B(\mathbf{w}^*, \epsilon)$ for some $\epsilon > 0$. Recall that for a pairwise potential function is said to be submodular if $(\phi_{uv}(1,1;\mathbf{x},\mathbf{w}^*) + \phi_{uv}(0,0;\mathbf{x},\mathbf{w}^*)) - (\phi_{uv}(1,0;\mathbf{x},\mathbf{w}^*) + \phi_{uv}(0,1;\mathbf{x},\mathbf{w}^*)) > 0$ and supermodular if $(\phi_{uv}(1,1;\mathbf{x},\mathbf{w}^*) + \phi_{uv}(0,0;\mathbf{x},\mathbf{w}^*)) - (\phi_{uv}(1,0;\mathbf{x},\mathbf{w}^*) + \phi_{uv}(0,1;\mathbf{x},\mathbf{w}^*)) < 0$. Also, recall Assumption 2: $\forall \mathbf{w}^* \in W^*$, we know that all ϕ_{uv} are either submodular or supermodular; moreover, we know if any given ϕ_{uv} is submodular or supermodular Since the above inequalities are strict, it is easy to see that if we pick a small enough ϵ, $\mathbf{w}$ also satisfies these inequalities.

Now consider $(\mathbf{x^j}, \mathbf{y^j}) \in D$ and let $E^j = \{(u,v) \in E | \triangle\phi_{uv} > 0, y^j_u = y^j_v \text{ or } \triangle\phi_{uv} < 0, y^j_u \neq y^j_v\}$. The corresponding decomposition is $\mathcal{S}_{pair}(\mathbf{y^j}) = \{c_1, \ldots, c_l\}$ where $c_1, \ldots, c_l$ correspond to the indices of the connected componenets in E^j. Let $E^{-j} = E \setminus E^j$.

Consider a $\mathbf{y} \in \mathcal{Y}$. For $k = 1, \ldots, l$, define $\mathbf{y}[\mathbf{k}] = (\mathbf{y}_{c_k}, \mathbf{y^j}_{-c_k})$ that is $\mathbf{y}[\mathbf{k}]$ is produced by replacing all labels of $\mathbf{y^j}$ listed in c_k by corresponding ones from $\mathbf{y}$; clearly $\mathbf{y}[\mathbf{k}] \in nbr(\mathbf{y^j})$.

Cor. 1 requires us to show that $\sum_{k=1}^{l} \big(f(\mathbf{y}[\mathbf{k}], x; w) - f(\mathbf{y^j}, \mathbf{x^j}; \mathbf{w})\big) \geq f(\mathbf{y}, \mathbf{x^j}; \mathbf{w}) - f(\mathbf{y^j}, \mathbf{x^j}; \mathbf{w})$. In order to show this, consider

$$
\begin{aligned}
& \left(\sum_{k=1}^{l} \big(f(\mathbf{y}[\mathbf{k}], \mathbf{x^j}; w) - f(\mathbf{y^j}, \mathbf{x^j}; \mathbf{w})\big)\right) \\
& - \big(f(\mathbf{y}, \mathbf{x^j}; \mathbf{w}) - f(\mathbf{y^j}, \mathbf{x^j}; \mathbf{w})\big) \\
= \;& 2 \sum_{(u,v) \in E^{-j}} \Big(\phi_{uv}(y^j_u, y_v; \mathbf{w}) + \phi_{uv}(y_u, y^j_v; \mathbf{w}) \\
& -\phi_{uv}(y_u, y_v; \mathbf{w}) - \phi_{uv}(y^j_u, y^j_v; \mathbf{w})\Big) \ .
\end{aligned}
$$

Recall that for any $(u,v) \in E^{-j}$, we have $\{\triangle\phi_{uv} \leq 0, y^j_u = y^j_v \text{ or } \triangle\,\phi_{uv} \geq 0, y^j_u \neq y^j_v\}$. Thus for each edge $e = (u,v) \in E^{-j}$, it is easy to see after exhaustively checking all 8 possible assignments to y_u, y_v, y^j_u, and y^j_v, that $\phi_{uv}(y^j_u, y_v; \mathbf{w}) + \phi_{uv}(y_u, y^j_v; \mathbf{w}) \geq \phi_{uv}(y_u, y_v; \mathbf{w}) + \phi_{uv}(y^j_u, y^j_v; \mathbf{w})$. Hence the R.H.S. above is non-negative. Thus for $\mathbf{w}$, we have

$$
\begin{aligned}
\textstyle\sum_{k=1}^{l} & \big(f(\mathbf{y}[\mathbf{k}], \mathbf{x^j}; \mathbf{w}) - f(\mathbf{y^j}, \mathbf{x^j}; \mathbf{w})\big) \\
& \geq f(\mathbf{y}, \mathbf{x^j}; \mathbf{w}) - f(\mathbf{y^j}, \mathbf{x^j}; \mathbf{w}) \ .
\end{aligned}
$$

which is the required condition in Cor. 1. □