Definition 2.10. A local search problem \( \Pi \) is PLS-complete if

- \( \Pi \) belongs to the complexity class PLS;
- every problem in PLS is PLS-reducible to \( \Pi \).

The above definitions imply the following: If there is a polynomial time algorithm that computes a local optimum for a PLS-complete problem \( \Pi \), then there exists a polynomial time algorithm for finding a local optimum for every problem in PLS. This holds since every problem in PLS can be reduced to \( \Pi \) in polynomial time (see Figure 8) and the reduction preserves local optima.

We will show that computing a Nash equilibrium for exact potential games is PLS-complete. We do so by a reduction from the weighted satisfiability problem.

Example 2.3. The weighted satisfiability problem is given as follows. We are given a formula in conjunctive normal form:

\[
 f = C_1 \land C_2 \land \cdots \land C_k,
\]

where each clause \( C_j, j \in [k], \) is a disjunction of literals. (Example: \( (y_1 \lor y_2) \land (\bar{y}_1 \lor y_3) \).) We assume that \( f \) consists of \( n \) variables \( y_1, \ldots, y_n \). Every clause \( C_j, j \in [k], \) has a non-negative weight \( w_j \). A feasible solution is an assignment \( y = (y_1, \ldots, y_n) \in \{0,1\}^n \) of 0/1-values to the variables. The total cost \( c(y) \) of an assignment \( y \) is given by the sum of the weights of the clauses that are false with respect to \( y \). Define the neighborhood of an assignment \( y \) as the set of assignments that are obtainable from \( y \) by changing a single variable \( y_i, i \in [n], \) from 0 to 1 or vice versa. The problem is to determine an assignment \( y \) that is a local minimum with respect to \( c \).

Clearly, the weighted satisfiability problem belongs to PLS. Moreover, the problem is PLS-complete.

Theorem 2.7. The problem of finding a pure Nash equilibrium for exact potential games is PLS-complete.

Proof. We observed above that the problem belongs to PLS. We reduce the weighted satisfiability problem to the problem of finding a Nash equilibrium in an exact potential game.

Consider an instance of the weighted satisfiability problem

\[
 f = C_1 \land C_2 \land \cdots \land C_k
\]
with \(n\) variables \(y_1, \ldots, y_n\) and weight \(w_j\) for clause \(C_j, j \in [k]\).

We derive a strategic game \(\Gamma = (N, (X_i)_{i \in N}, (u_i)_{i \in N})\) from this instance as follows: We associate a player \(i \in N := [n]\) with every variable \(y_i\) of \(f\). Imagine that every clause \(C_j, j \in [k]\), corresponds to a unique resource \(j\) and that each player can allocate certain subsets of these resources: For player \(i \in N\), define

\[
J(i) := \{ j \in [k] : y_i \text{ occurs in } C_j \} \quad \text{and} \quad \bar{J}(i) := \{ j \in [k] : \bar{y}_i \text{ occurs in } C_j \}
\]
as the resource sets that correspond to clauses that contain the literals \(y_i\) and \(\bar{y}_i\), respectively. The strategy set \(X_i\) of player \(i \in N\) consists of two strategies: \(X_i = \{ J(i), \bar{J}(i) \}\). Our interpretation will be that \(y_i = 0\) if player \(i \in N\) chooses strategy \(x_i = J(i)\), while \(y_i = 1\) if player \(i \in N\) chooses \(x_i = \bar{J}(i)\).

The crucial observation is that a clause \(C_j, j \in [k]\), with \(l_j\) literals is false iff there are exactly \(l_j\) players that have chosen resource \(j\). For a given strategy profile \(x \in X\), let \(n_j(x)\) refer to the number of players that have chosen resource \(j \in [k]\), i.e.,

\[
n_j(x) := |\{i \in N : j \in x_i\}|.
\]

Assign every resource \(j \in [k]\) a cost \(c_j(x)\) as follows:

\[
c_j(x) := \begin{cases} 0 & \text{if } n_j(x) < l_j \\ w_j & \text{otherwise.} \end{cases}
\]

Each player’s goal is to minimize the total cost of the resources he allocates. That is, the utility \(u_i(x)\) of player \(i \in N\) (which he wants to maximize) is defined as \(u_i(x) := -\sum_{j \in x_i} c_j(x)\). This reduction can be done in polynomial time. Moreover, every assignment of the weighted satisfiability problem can be mapped in polynomial time to a corresponding strategy profile of the resulting strategic game \(\Gamma\) and vice versa.

We show that

\[
\Phi(x) := \sum_{j \in [k]} c_j(x)
\]
is an exact potential function for \(\Gamma\). To see this note that \(\Phi(x)\) is equal the total cost \(c(y)\) of the corresponding variable assignment \(y\). Moreover, \(-u_i(x)\) accounts for the total cost of all false clauses that contain variable \(y_i\) (either negated or not). If player \(i\) changes his strategy from \(x_i\) to \(x'_i\) we therefore have:

\[
u_i(x_{-i}, x'_i) - u_i(x_{-i}, x_i) = \Phi(x_{-i}, x_i) - \Phi(x_{-i}, x'_i).
\]

That is, \(\Phi\) is an exact potential function. By Theorem 2.2, the local minima of \(\Phi\) correspond exactly to the set of pure Nash equilibria of \(\Gamma\). The described reduction preserves local optima and thus all conditions of Definition 2.9 are met. This concludes the proof. \(\square\)