

```
In [112... import os
import json
import math
import numpy as np
import pandas as pd
import plotly.io as pio
import plotly.express as px
import plotly.graph_objects as go
from collections import defaultdict
from plotly.subplots import make_subplots
pio.kaleido.scope.mathjax = None
pd.options.display.max_rows = None
```

```
In [113... ALGOS = ["bc_beta0.0", "bc_beta0.5", "bc_beta1.0", "omap1", "indd", "vdn", "miso-sl", "miso-mini", "miso_alpha0.00", "miso_alp
MAIN_ALGOS = ["bc_beta0.0", "bc_beta0.5", "bc_beta1.0", "omap1", "indd", "vdn", "miso-sl", "miso_alpha0.05"]
MAIN_ALGOS_2 = ["bc_beta0.0", "bc_beta0.5", "bc_beta1.0", "indd", "vdn", "miso-sl", "miso_alpha0.05"]
ABLATION_ALGOS = ["miso_alpha0.00", "miso_alpha0.05", "miso_alpha0.10", "miso_alpha0.50", "miso_alpha1.00", "miso_alpha10.00"]
SMACV1_ENV_NAMES = ["2c_vs_64zg", "5m_vs_6m", "6h_vs_8z", "corridor"]
SMACV2_ENV_NAMES = [f"{map_name}_{map_mode}" for map_name in ["protoss", "terran", "zerg"] for map_mode in ["5_vs_5", "10_vs_1
MAMUJOCO_ENV_NAMES = ["Hopper-v2", "Ant-v2", "HalfCheetah-v2"]
RENAME_ALGOS = {
    "bc_beta0.0": "BC ( $\beta=0.0$ )",
    "bc_beta0.5": "BC ( $\beta=0.5$ )",
    "bc_beta1.0": "BC ( $\beta=1.0$ )",
    "omap1": "OMAPL",
    "indd": "INDD",
    "vdn": "VDN",
    "miso-sl": "MARL-SL",
    "miso-mini": "MisoDICE (4o-mini)",
    "miso_alpha0.00": "MisoDICE ( $\alpha=0.00$ )",
    "miso_alpha0.05": "MisoDICE (ours)",
    "miso_alpha0.10": "MisoDICE ( $\alpha=0.10$ )",
    "miso_alpha0.50": "MisoDICE ( $\alpha=0.50$ )",
    "miso_alpha1.00": "MisoDICE ( $\alpha=1.00$ )",
    "miso_alpha10.00": "MisoDICE ( $\alpha=10.00$ )",
}
colors = px.colors.qualitative.Plotly
COLOR_MAPS = {
    "bc_beta0.0": colors[4],
    "bc_beta0.5": colors[9],
    "bc_beta1.0": colors[2],
    "omap1": colors[7],
    "indd": colors[3],
    "vdn": colors[0],
    "miso-sl": colors[8],
    "miso-mini": colors[0],
    "miso_alpha0.00": colors[5],
    "miso_alpha0.05": colors[1],
    "miso_alpha0.10": colors[6],
    "miso_alpha0.50": colors[7],
    "miso_alpha1.00": colors[8],
    "miso_alpha10.00": "#feb406",

    50: colors[0],
    200: colors[1],
    400: colors[2],
    800: colors[3],
    1200: colors[4],
}

LOG_DIR = "logs"
```

```
In [114... def darken_color(hex_color, factor=0.92):
    r = int(hex_color[1:3], 16)
    g = int(hex_color[3:5], 16)
    b = int(hex_color[5:7], 16)

    r = int(r * factor)
    g = int(g * factor)
    b = int(b * factor)

    r = max(0, min(r, 255))
    g = max(0, min(g, 255))
    b = max(0, min(b, 255))

    return "#{:02x}{:02x}{:02x}".format(r, g, b)
```

```
In [115... COLOR_MAPS = {k: darken_color(v) for k, v in COLOR_MAPS.items()}
```

```
In [116... def load_results(path):
    if not os.path.exists(path):
        return {}
    with open(path, "r") as f:
```

```
data = json.load(f)
return data
```

```
In [117... def smooth(scalars, weight=0.75):
    last = 0
    smoothed = []
    for num_acc, next_val in enumerate(scalars):
        last = last * weight + (1 - weight) * next_val
        smoothed.append(last / (1 - math.pow(weight, num_acc+1)))
    return smoothed
```

```
In [118... def load_data(use_llm=False, exsize=200, algos=None):
    data_returns = defaultdict(lambda: defaultdict(lambda: defaultdict(list)))
    data_winrates = defaultdict(lambda: defaultdict(lambda: defaultdict(list)))
    if not algos:
        algos = ALGOS
    for algo in algos:
        for env_name in MAMUJOCO_ENV_NAMES + SMACV1_ENV_NAMES + SMACV2_ENV_NAMES:
            for seed in range(4):
                if use_llm:
                    if env_name in MAMUJOCO_ENV_NAMES:
                        continue
                    path = f"{LOG_DIR}/{algo}/{env_name}_llm/seed{seed}/exsize{exsize}/results.json"
                else:
                    path = f"{LOG_DIR}/{algo}/{env_name}/seed{seed}/exsize{exsize}/results.json"

                if algo == "omap1":
                    path = f"{LOG_DIR}/{algo}/{env_name}_llm/seed{seed}/results.json"

                data = load_results(path)
                for step in range(101):
                    result = data[f"step_{step}"] if algo != "omap1" else data
                    if "returns" in result:
                        data_returns[algo][env_name][step].append(np.mean(result["returns"]))
                    if "winrates" in result:
                        data_winrates[algo][env_name][step].append(np.mean(result["winrates"]))
    return data_returns, data_winrates
```

```
In [119... def analyze_data(data, tag="returns"):
    if isinstance(data, float):
        return "NaN"
    items = [data[step] for step in sorted(data.keys())]
    items = np.array(items)
    items = [smooth(items[:, i]) for i in range(items.shape[-1])]
    items = np.array(items)
    items = items[:, -1]
    mean = np.mean(items)
    std = np.std(items)
    if tag == "returns":
        return f"{mean:.1f} ± {std:.1f}"
    elif tag == "winrates":
        return f"{100*mean:.1f} ± {100*std:.1f}"
    else:
        return items
```

```
In [120... data_returns_llm, data_winrates_llm = load_data(use_llm=True, algos=MAIN_ALGOS)
pd_returns_llm = pd.DataFrame.from_dict(data_returns_llm).rename(columns=RENAME_ALGOS)
pd_winrates_llm = pd.DataFrame.from_dict(data_winrates_llm).rename(columns=RENAME_ALGOS)
```

```
In [121... pd_returns_llm.map(analyze_data)
```

Out[121...

	BC ($\beta=0.0$)	BC ($\beta=0.5$)	BC ($\beta=1.0$)	OMAPL	INDD	VDN	MARL-SL	MisoDICE (ours)
2c_vs_64zg	8.5 ± 0.1	9.7 ± 0.3	12.6 ± 0.3	12.2 ± 0.4	14.6 ± 1.0	14.0 ± 1.6	12.7 ± 0.6	16.4 ± 1.3
5m_vs_6m	5.0 ± 1.1	6.7 ± 0.0	6.1 ± 0.1	5.7 ± 0.2	6.7 ± 0.1	6.8 ± 0.1	6.2 ± 1.4	7.3 ± 0.1
6h_vs_8z	7.0 ± 0.0	7.4 ± 0.0	7.2 ± 0.1	6.6 ± 0.2	7.5 ± 0.2	7.8 ± 0.1	8.2 ± 0.2	8.7 ± 0.2
corridor	1.5 ± 0.1	1.5 ± 0.2	4.3 ± 0.7	2.2 ± 1.3	4.4 ± 1.2	1.8 ± 0.2	4.7 ± 0.6	5.8 ± 0.8
protoss_5_vs_5	9.2 ± 0.1	11.7 ± 0.5	10.2 ± 0.5	9.6 ± 1.1	10.9 ± 0.1	11.6 ± 0.3	11.5 ± 0.2	12.4 ± 0.5
protoss_10_vs_10	10.3 ± 0.6	11.8 ± 0.5	10.6 ± 0.2	10.1 ± 0.9	11.0 ± 0.7	11.9 ± 0.4	12.4 ± 0.2	12.9 ± 0.2
protoss_10_vs_11	8.2 ± 0.4	9.6 ± 0.4	8.7 ± 0.3	8.5 ± 1.2	9.4 ± 0.4	9.9 ± 0.3	10.4 ± 0.1	10.7 ± 0.4
protoss_20_vs_20	10.1 ± 0.2	10.4 ± 0.5	10.5 ± 0.3	9.4 ± 0.4	11.4 ± 0.5	13.1 ± 0.4	12.1 ± 0.5	13.5 ± 0.5
protoss_20_vs_23	8.1 ± 0.2	8.6 ± 0.3	8.3 ± 0.2	7.9 ± 0.3	9.6 ± 0.3	9.6 ± 0.3	10.3 ± 0.4	10.6 ± 0.2
terran_5_vs_5	6.5 ± 0.8	8.1 ± 0.5	7.1 ± 0.6	6.2 ± 0.6	7.9 ± 0.5	8.1 ± 0.4	8.3 ± 1.0	9.1 ± 0.3
terran_10_vs_10	6.6 ± 0.3	7.4 ± 0.4	6.7 ± 0.6	6.9 ± 1.1	7.6 ± 0.4	7.7 ± 0.2	8.0 ± 0.4	9.1 ± 1.3
terran_10_vs_11	4.7 ± 0.2	5.7 ± 0.3	5.2 ± 0.3	4.2 ± 0.6	5.7 ± 0.5	5.7 ± 0.4	6.0 ± 0.2	6.4 ± 0.5
terran_20_vs_20	6.9 ± 0.4	7.9 ± 0.8	6.7 ± 0.2	6.9 ± 0.5	8.0 ± 0.5	8.6 ± 0.3	8.2 ± 0.5	9.2 ± 0.6
terran_20_vs_23	4.0 ± 0.3	5.1 ± 0.4	4.3 ± 0.3	4.3 ± 0.4	5.1 ± 0.4	5.1 ± 0.6	5.6 ± 0.3	5.6 ± 0.4
zerg_5_vs_5	5.7 ± 0.5	6.6 ± 0.4	5.9 ± 0.3	6.1 ± 0.5	6.4 ± 0.2	7.1 ± 0.5	7.1 ± 0.9	7.5 ± 0.1
zerg_10_vs_10	7.3 ± 0.1	8.7 ± 0.6	7.4 ± 0.7	6.8 ± 0.6	8.2 ± 0.2	9.0 ± 0.4	9.7 ± 0.5	10.2 ± 0.6
zerg_10_vs_11	7.3 ± 0.2	8.3 ± 0.4	7.3 ± 0.5	7.2 ± 0.4	8.0 ± 0.2	8.8 ± 0.4	9.1 ± 0.2	9.4 ± 0.3
zerg_20_vs_20	7.4 ± 0.6	9.0 ± 0.5	7.7 ± 0.2	6.9 ± 0.5	8.3 ± 0.4	8.8 ± 0.6	9.0 ± 0.5	10.2 ± 0.6
zerg_20_vs_23	7.1 ± 0.3	7.9 ± 0.3	7.0 ± 0.2	7.1 ± 0.4	8.2 ± 0.4	8.8 ± 0.2	8.7 ± 0.5	9.5 ± 0.2

In [122...

```
pd_winrates_llm.map(lambda x: analyze_data(x, tag="winrates"))
```

Out[122...

	BC ($\beta=0.0$)	BC ($\beta=0.5$)	BC ($\beta=1.0$)	OMAPL	INDD	VDN	MARL-SL	MisoDICE (ours)
2c_vs_64zg	0.2 ± 0.2	0.5 ± 0.3	8.9 ± 2.9	3.9 ± 3.4	11.7 ± 5.5	10.6 ± 6.0	2.7 ± 1.5	13.0 ± 9.0
5m_vs_6m	0.2 ± 0.4	0.9 ± 0.6	0.1 ± 0.1	0.0 ± 0.0	0.2 ± 0.2	1.1 ± 0.8	0.9 ± 0.9	1.2 ± 0.5
6h_vs_8z	0.2 ± 0.2	0.0 ± 0.0	0.2 ± 0.3	0.0 ± 0.0	0.1 ± 0.1	1.0 ± 0.6	1.2 ± 0.1	1.1 ± 0.8
corridor	0.1 ± 0.1	0.6 ± 0.7	0.3 ± 0.4	0.0 ± 0.0	0.1 ± 0.1	0.9 ± 0.6	0.7 ± 0.7	1.4 ± 0.6
protoss_5_vs_5	13.8 ± 2.7	17.5 ± 3.8	14.2 ± 2.4	14.1 ± 10.2	12.4 ± 3.1	15.6 ± 4.5	10.8 ± 1.6	20.7 ± 0.9
protoss_10_vs_10	12.1 ± 2.3	12.7 ± 0.9	11.3 ± 2.9	11.7 ± 3.4	8.9 ± 1.4	11.8 ± 4.3	9.5 ± 1.7	14.1 ± 2.1
protoss_10_vs_11	2.1 ± 0.9	3.5 ± 2.0	1.8 ± 0.6	0.8 ± 1.4	2.0 ± 0.4	3.5 ± 0.3	2.9 ± 0.9	4.7 ± 0.3
protoss_20_vs_20	4.5 ± 1.8	3.4 ± 1.9	7.0 ± 2.5	3.1 ± 3.8	5.2 ± 1.3	8.6 ± 2.0	5.2 ± 1.9	11.0 ± 3.2
protoss_20_vs_23	1.5 ± 0.8	0.8 ± 0.4	1.8 ± 0.8	0.8 ± 1.4	2.4 ± 0.9	2.0 ± 0.9	2.6 ± 1.0	3.8 ± 2.0
terran_5_vs_5	10.1 ± 2.3	12.8 ± 1.3	13.7 ± 3.5	10.2 ± 4.6	10.6 ± 1.3	9.7 ± 1.9	10.4 ± 3.2	14.2 ± 3.1
terran_10_vs_10	7.3 ± 2.0	7.7 ± 2.2	7.9 ± 2.4	9.4 ± 3.8	8.0 ± 2.4	8.2 ± 1.6	7.0 ± 1.9	12.0 ± 1.7
terran_10_vs_11	1.9 ± 0.8	3.1 ± 1.4	2.5 ± 1.1	0.8 ± 1.4	1.8 ± 0.5	2.6 ± 1.2	2.4 ± 0.8	4.2 ± 1.6
terran_20_vs_20	4.5 ± 1.3	7.4 ± 2.5	4.4 ± 0.6	4.7 ± 3.5	5.8 ± 0.7	7.1 ± 1.4	4.5 ± 1.5	8.8 ± 1.5
terran_20_vs_23	0.6 ± 1.0	1.2 ± 0.9	0.7 ± 0.7	0.0 ± 0.0	0.8 ± 0.4	1.3 ± 0.7	1.3 ± 1.2	1.8 ± 1.4
zerg_5_vs_5	6.2 ± 0.8	5.6 ± 1.1	5.9 ± 0.5	6.2 ± 5.8	6.0 ± 1.4	6.7 ± 1.1	7.0 ± 1.8	7.9 ± 1.0
zerg_10_vs_10	4.5 ± 1.4	6.9 ± 1.8	5.0 ± 1.8	3.9 ± 3.4	4.6 ± 1.0	5.7 ± 1.2	5.2 ± 0.6	8.9 ± 2.1
zerg_10_vs_11	5.7 ± 1.9	5.8 ± 1.9	5.2 ± 1.3	2.3 ± 1.4	4.7 ± 1.6	6.0 ± 1.9	3.4 ± 0.7	6.8 ± 1.1
zerg_20_vs_20	0.8 ± 0.9	1.5 ± 0.6	1.3 ± 0.8	0.0 ± 0.0	0.2 ± 0.2	1.2 ± 0.5	0.6 ± 0.2	2.4 ± 0.6
zerg_20_vs_23	1.2 ± 0.9	0.7 ± 0.3	1.2 ± 0.6	1.6 ± 1.6	1.4 ± 0.5	1.9 ± 0.9	1.8 ± 1.1	2.7 ± 0.4

In [123...

```
def update_legend(fig, tag="returns", distance=1.1, yrange=None):
    trace_names = []
    for trace in fig.data:
        if trace.name is not None and trace.name not in trace_names:
            trace_names.append(trace.name)
            trace.update(showlegend=True)
        else:
            trace.update(showlegend=False)
    fig.update_layout(legend=dict(orientation="h", yanchor="bottom", y=distance, xanchor="right", x=1))

    fig.update_xaxes(showgrid=True)
    fig.update_yaxes(showgrid=True)
```

```

fig.update_xaxes(range=[0, 100], dtick=50, minor=dict(ticklen=3, nticks=3))

if tag == "winrates":
    fig.update_yaxes(tickformat=".0%", dtick=0.1, minor=dict(ticklen=3, nticks=2))
else:
    fig.update_yaxes(tickformat="~s")
if yrange is not None:
    fig.update_yaxes(range=yrange)
return fig

```

In [124...

```

def create_scatters(env_name, data_dict, y_range=None, tag="returns", algos=None, rename_algos=None):
    fig = go.Figure()
    if not algos:
        algos = MAIN_ALGOS_2
    if not rename_algos:
        rename_algos = RENAME_ALGOS
    for algo in algos:
        steps = []
        values = []
        stds = []
        for step in range(101):
            if step < 1:
                continue
            data = np.array(data_dict[algo][env_name][step])
            steps.append(step)
            values.append(data.mean())
            stds.append(data.std())
        values = smooth(values)

        uppers = [value + std for value, std in zip(values, stds)]
        lowers = [value - std for value, std in zip(values, stds)]

        if tag == "winrates":
            uppers = [min(1.0, value) for value in uppers]
            lowers = [max(0.0, value) for value in lowers]

        color = COLOR_MAPS.get(algo, colors[2])
        algo_name = rename_algos.get(algo, algo)
        opacity = 0.1
        line_width = 1.5
        fig.add_trace(go.Scatter(x=steps, y=values, mode="lines", name=algo_name, line_color=color, line_width=line_width))
        fig.add_trace(go.Scatter(x=steps+steps[::-1], y=uppers+lowers[::-1], fill="toself", fillcolor=color, line_color=color,

    if tag == "winrates":
        tickformat = ".0%"
    else:
        tickformat = "~s"

    fig.update_layout(template='simple_white', margin=dict(l=0, r=0, t=0, b=0, pad=0, autoexpand=True))
    fig.update_layout(height=130, width=180)
    fig.update_xaxes(range=[0, 100], dtick=50, minor=dict(ticklen=3, nticks=4))
    fig.update_yaxes(range=y_range, tickformat=tickformat)
    fig.update_layout(showlegend=False)
    return fig

```

In [125...

```

def show_fig(plotly_figs, env_names, n_rows, n_cols, height, width, title):
    tag = "returns" if "returns" in title.lower() else "winrates"
    prefix = "ablation_" if "ablation" in title.lower() else ""
    fig = make_subplots(rows=n_rows, cols=n_cols, subplot_titles=env_names)
    if any(env_name in SMACV2_ENV_NAMES for env_name in env_names):
        for i, mode in enumerate(["5_vs_5", "10_vs_10", "10_vs_11", "20_vs_20", "20_vs_23"]):
            for j, env_name in enumerate(["protoss", "terran", "zerg"]):
                plotly_fig = plotly_figs[f"{env_name}_{mode}"]
                plotly_fig.write_image(f"graphs/{prefix}{env_name}_{mode}_llm_{tag}.pdf")
                fig.add_traces(plotly_fig.data, rows=j+1, cols=i+1)
    else:
        for i, env_name in enumerate(env_names):
            plotly_fig = plotly_figs[env_name]
            plotly_fig.write_image(f"graphs/{prefix}{env_name}_llm_{tag}.pdf")
            fig.add_traces(plotly_fig.data, rows=1, cols=i+1)
    fig.update_layout(template='simple_white')
    fig.update_layout(height=height, width=width, title_text=title)
    fig = update_legend(fig, tag=tag, distance=1.15)
    fig.show("svg")
    fig.update_layout(title_text=None)
    title = title.replace(" - ", "_").replace(" ", "_").replace(":", "_").replace("__", "_")
    print(f"Saving {title}.pdf")
    fig.write_image(f"graphs/{title}.pdf")

```

In [126...

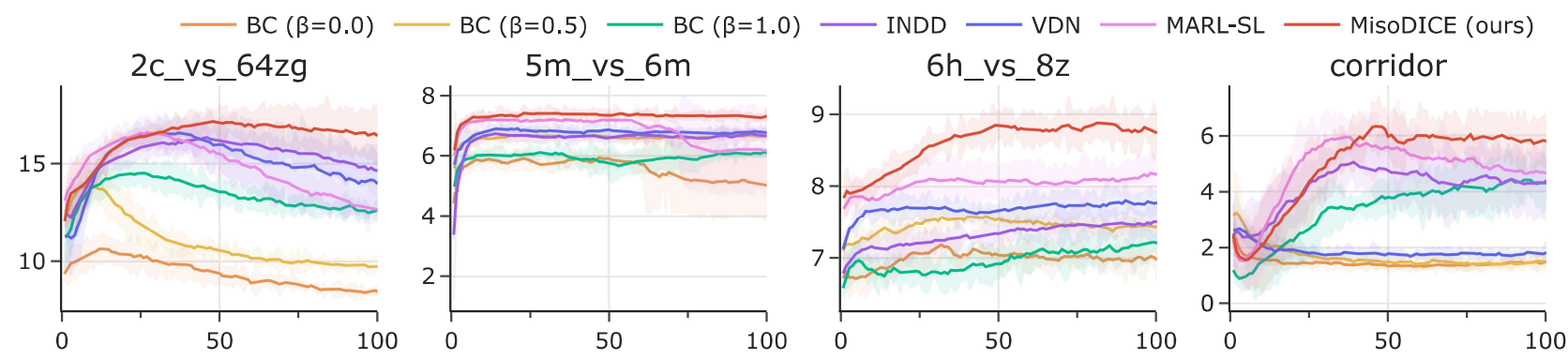
```

plotly_figs = {}
for env_name in SMACV1_ENV_NAMES + SMACV2_ENV_NAMES:
    plotly_figs[env_name] = create_scatters(env_name, data_returns_llm)

show_fig(plotly_figs, SMACV1_ENV_NAMES, n_rows=1, n_cols=4, height=300, width=950, title="SMACv1 - Returns - LLM-based")
show_fig(plotly_figs, SMACV2_ENV_NAMES, n_rows=3, n_cols=5, height=640, width=1200, title="SMACv2 - Returns - LLM-based")

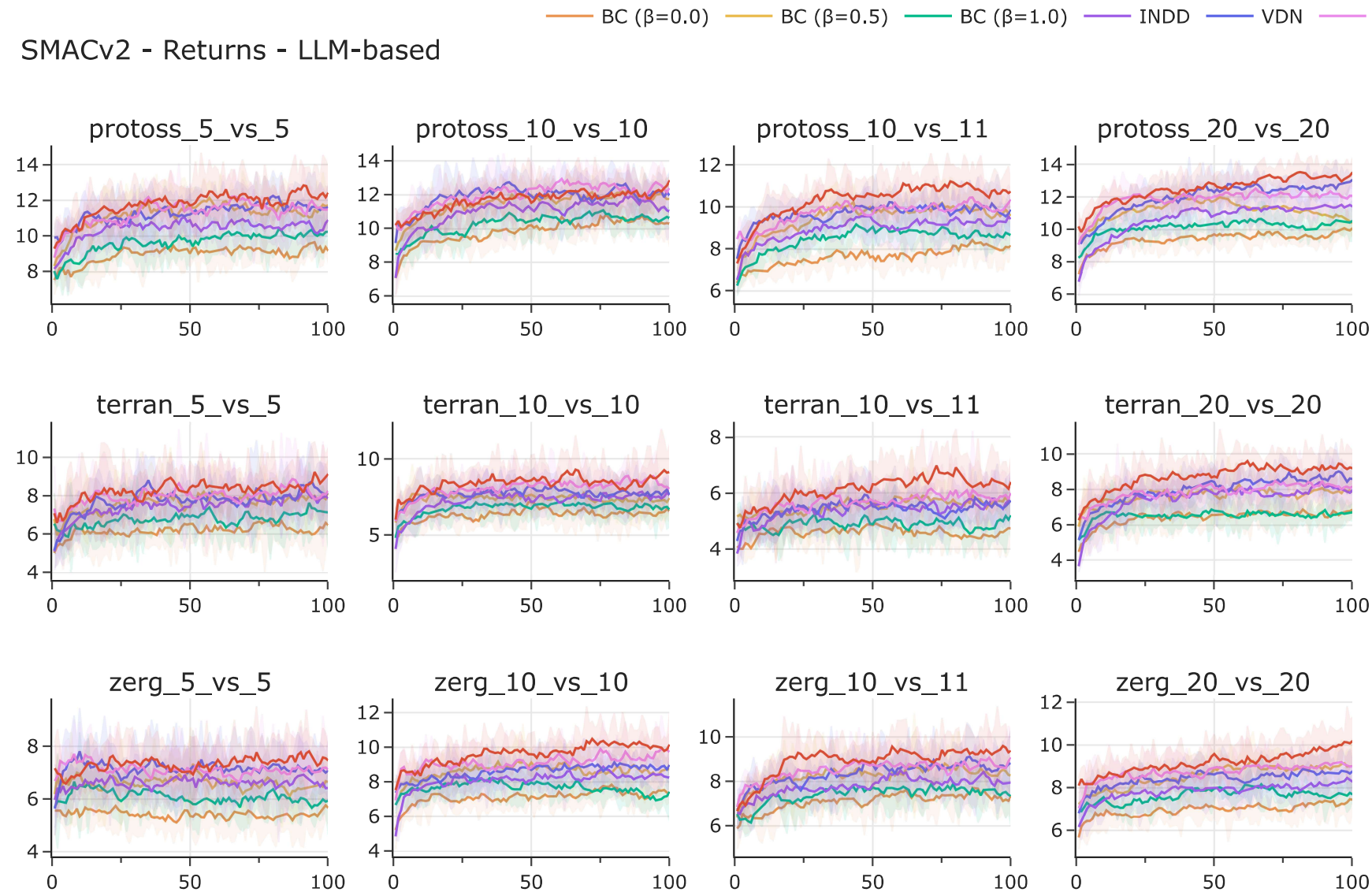
```


SMACv1 - Returns - LLM-based



Saving SMACv1_Returns_LLM-based.pdf

SMACv2 - Returns - LLM-based



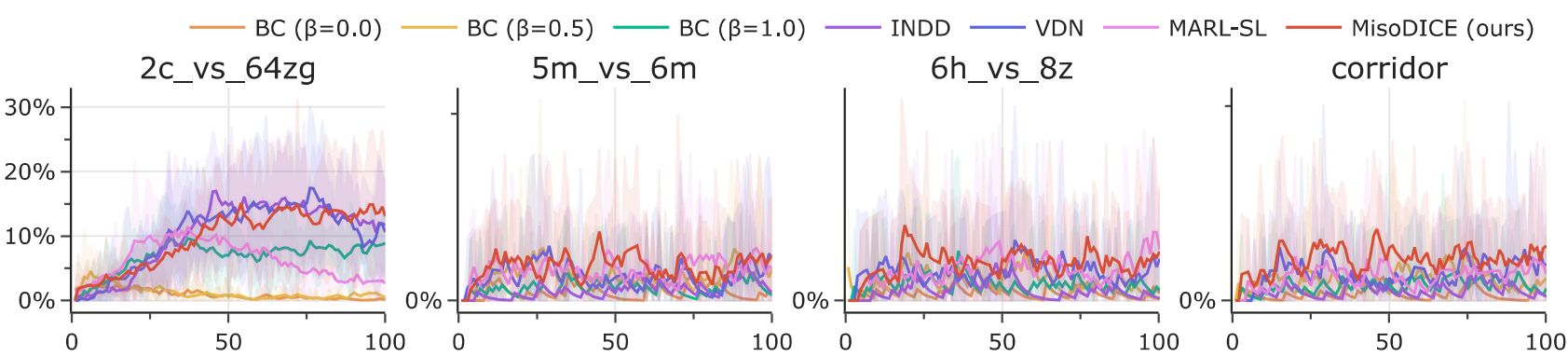
Saving SMACv2_Returns_LLM-based.pdf

In [127...

```
plotly_figs = {}
for env_name in SMACV1_ENV_NAMES + SMACV2_ENV_NAMES:
    plotly_figs[env_name] = create_scatters(env_name, data_winrates_llm, tag="winrates")

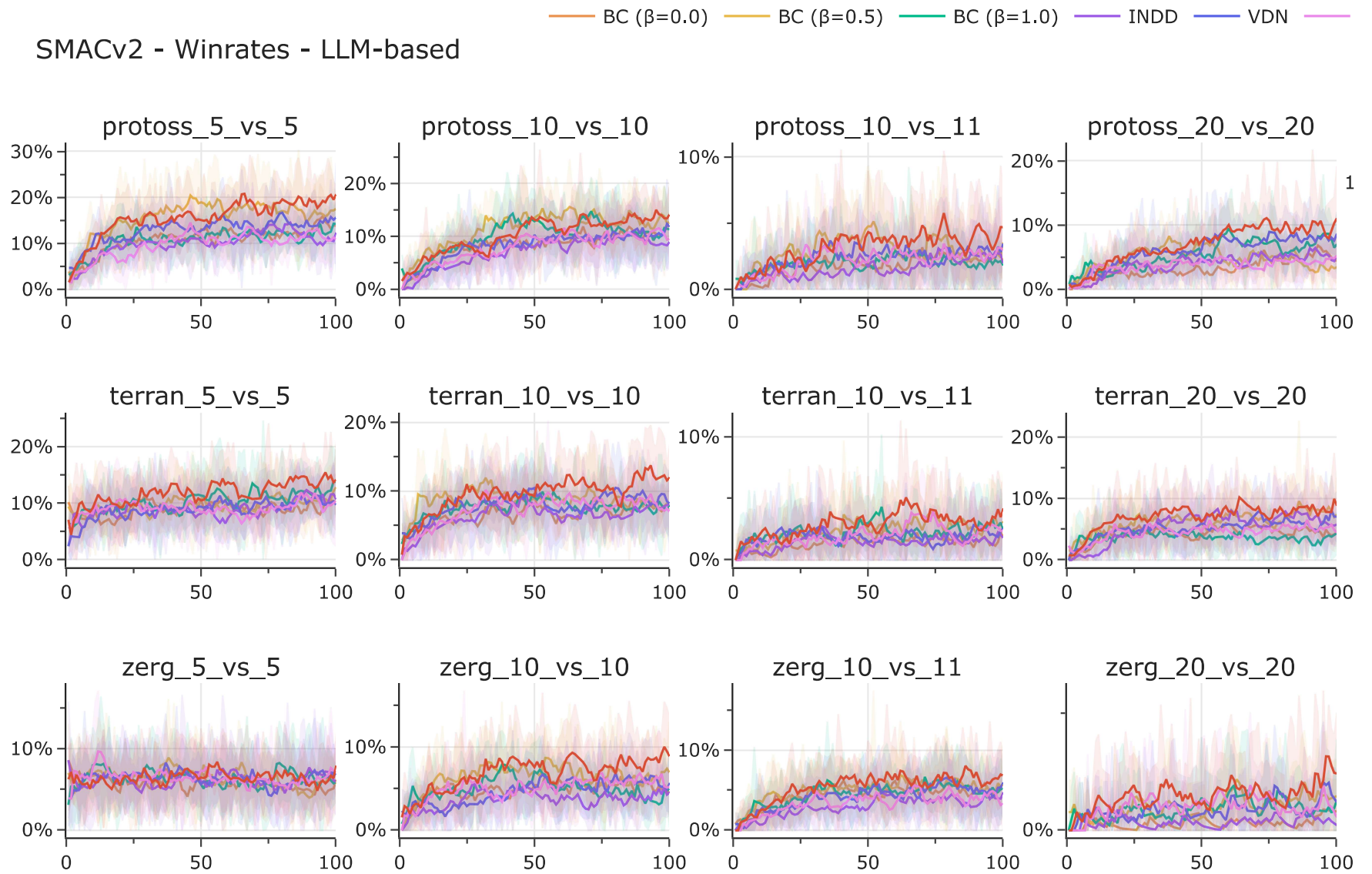
show_fig(plotly_figs, SMACV1_ENV_NAMES, n_rows=1, n_cols=4, height=300, width=950, title="SMACv1 - Winrates - LLM-based")
show_fig(plotly_figs, SMACV2_ENV_NAMES, n_rows=3, n_cols=5, height=640, width=1200, title="SMACv2 - Winrates - LLM-based")
```

SMACv1 - Winrates - LLM-based



Saving SMACv1_Winrates_LLM-based.pdf

SMACv2 - Winrates - LLM-based



Saving SMACv2_Winrates_LLM-based.pdf

```
In [128... def show_box_chart(pd_data, env_names, tag="returns"):
    pd_data = pd_data.map(lambda x: analyze_data(x, tag=None))
    fig = go.Figure()
    for i, exsize in enumerate([50, 200, 400, 800, 1200]):
        all_values = []
        for env_name in env_names:
            values = pd_data[exsize][env_name]
            all_values.extend(values)
        fig.add_trace(go.Box(y=all_values, name=exsize, marker_color=colors[0], boxpoints=False))

    if tag == "winrates":
        tickformat = ".0%"
    else:
        tickformat = "~s"

    fig.update_layout(template='simple_white', margin=dict(l=0, r=0, t=0, b=0, pad=0, autoexpand=True))
    fig.update_layout(height=160, width=240)
    fig.update_layout(showlegend=False)
    fig.update_yaxes(tickformat=tickformat)
    env_name = env_name.split("_")[0]
    fig.write_image(f"graphs/{env_name}_llm_{tag}_box.pdf")
    fig.show("svg")
```

```
In [129... ablation_exsize_returns_llm = {}
ablation_exsize_winrates_llm = {}
for exsize in [50, 200, 400, 800, 1200]:
    data_returns, data_winrates = load_data(use_llm=True, exsize=exsize, algos=["miso_alpha0.05"])
    ablation_exsize_returns_llm[exsize] = data_returns["miso_alpha0.05"]
    ablation_exsize_winrates_llm[exsize] = data_winrates["miso_alpha0.05"]
```

```
In [130... topk_names = {exsize: f"topk={exsize}" for exsize in [50, 200, 400, 800, 1200]}
PROTOSS_ENVS = [env_name for env_name in SMACV2_ENV_NAMES if "protoss" in env_name]
TERRAN_ENVS = [env_name for env_name in SMACV2_ENV_NAMES if "terran" in env_name]
ZERG_ENVS = [env_name for env_name in SMACV2_ENV_NAMES if "zerg" in env_name]
```

```
In [131... pd_returns_llm = pd.DataFrame.from_dict(ablation_exsize_returns_llm)
pd_returns_llm.rename(columns=topk_names).map(analyze_data)
```


Out[131...

	topk=50	topk=200	topk=400	topk=800	topk=1200
2c_vs_64zg	11.8 ± 1.4	16.4 ± 1.3	9.7 ± 0.1	10.8 ± 0.7	10.4 ± 0.5
5m_vs_6m	6.9 ± 0.4	7.3 ± 0.1	6.8 ± 0.2	6.5 ± 0.1	6.4 ± 0.2
6h_vs_8z	7.9 ± 0.1	8.7 ± 0.2	7.8 ± 0.2	7.5 ± 0.1	7.3 ± 0.1
corridor	3.1 ± 1.0	5.8 ± 0.8	1.9 ± 0.1	1.7 ± 0.2	1.7 ± 0.1
protoss_5_vs_5	11.9 ± 0.2	12.4 ± 0.5	11.3 ± 0.2	10.9 ± 0.2	10.9 ± 0.5
protoss_10_vs_10	11.8 ± 0.1	12.9 ± 0.2	12.1 ± 0.3	11.5 ± 0.3	11.2 ± 0.4
protoss_10_vs_11	9.6 ± 0.1	10.7 ± 0.4	9.5 ± 0.3	9.3 ± 0.5	9.2 ± 0.3
protoss_20_vs_20	11.4 ± 0.5	13.5 ± 0.5	12.1 ± 0.1	11.4 ± 0.3	11.2 ± 0.3
protoss_20_vs_23	9.8 ± 0.3	10.6 ± 0.2	10.0 ± 0.2	9.3 ± 0.2	8.7 ± 0.4
terran_5_vs_5	8.0 ± 0.3	9.1 ± 0.3	7.3 ± 0.4	7.7 ± 0.4	7.0 ± 0.5
terran_10_vs_10	8.4 ± 0.4	9.1 ± 1.3	8.1 ± 0.5	7.6 ± 0.7	7.0 ± 0.4
terran_10_vs_11	5.5 ± 0.1	6.4 ± 0.5	6.1 ± 0.2	5.7 ± 0.4	5.4 ± 0.5
terran_20_vs_20	8.0 ± 0.4	9.2 ± 0.6	8.2 ± 0.6	8.0 ± 0.5	7.9 ± 0.5
terran_20_vs_23	5.1 ± 0.2	5.6 ± 0.4	5.1 ± 0.2	5.2 ± 0.2	4.8 ± 0.2
zerg_5_vs_5	7.4 ± 0.4	7.5 ± 0.1	6.9 ± 0.5	6.4 ± 0.5	6.2 ± 0.5
zerg_10_vs_10	9.2 ± 0.3	10.2 ± 0.6	9.0 ± 0.3	8.8 ± 0.2	8.3 ± 0.2
zerg_10_vs_11	8.5 ± 0.4	9.4 ± 0.3	8.5 ± 0.0	8.2 ± 0.2	7.6 ± 0.3
zerg_20_vs_20	8.9 ± 0.1	10.2 ± 0.6	8.8 ± 0.9	8.3 ± 0.5	8.1 ± 0.6
zerg_20_vs_23	8.4 ± 0.4	9.5 ± 0.2	8.8 ± 0.2	8.5 ± 0.2	8.0 ± 0.1

In [132...

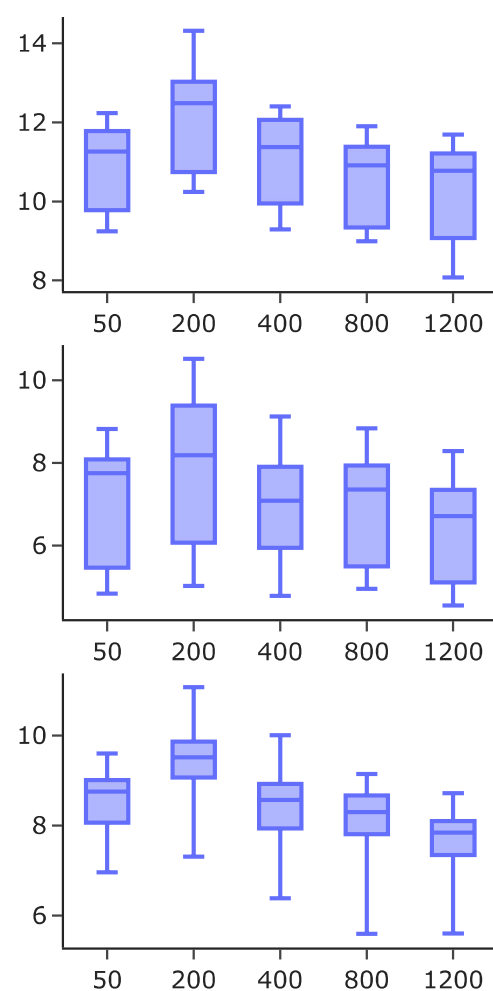
```
pd_winrates_llm = pd.DataFrame.from_dict(ablation_exsize_winrates_llm)
pd_winrates_llm.rename(columns=topk_names).map(lambda x: analyze_data(x, tag="winrates"))
```

Out[132...

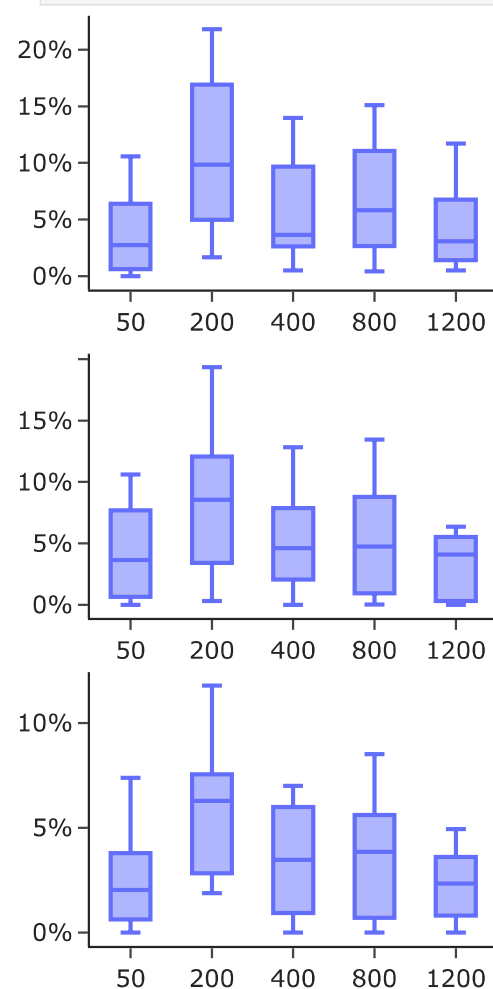
	topk=50	topk=200	topk=400	topk=800	topk=1200
2c_vs_64zg	5.6 ± 4.9	13.0 ± 9.0	1.6 ± 0.7	1.7 ± 0.8	0.6 ± 0.7
5m_vs_6m	0.0 ± 0.0	1.2 ± 0.5	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
6h_vs_8z	0.0 ± 0.0	1.1 ± 0.8	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
corridor	0.0 ± 0.0	1.4 ± 0.6	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0
protoss_5_vs_5	8.9 ± 1.9	20.7 ± 0.9	10.1 ± 2.8	12.5 ± 1.7	9.4 ± 1.7
protoss_10_vs_10	5.3 ± 1.8	14.1 ± 2.1	11.1 ± 3.3	10.0 ± 3.0	5.7 ± 1.6
protoss_10_vs_11	0.3 ± 0.4	4.7 ± 0.3	2.2 ± 1.2	2.0 ± 1.0	1.6 ± 0.6
protoss_20_vs_20	2.8 ± 0.8	11.0 ± 3.2	3.6 ± 1.4	6.1 ± 1.9	3.4 ± 0.9
protoss_20_vs_23	0.6 ± 0.4	3.8 ± 2.0	2.2 ± 0.8	1.8 ± 1.2	1.0 ± 0.4
terran_5_vs_5	8.5 ± 1.9	14.2 ± 3.1	9.8 ± 2.1	10.4 ± 2.1	5.1 ± 1.1
terran_10_vs_10	7.7 ± 0.6	12.0 ± 1.7	7.9 ± 2.1	8.1 ± 3.8	5.2 ± 0.3
terran_10_vs_11	0.8 ± 0.6	4.2 ± 1.6	2.2 ± 1.2	1.4 ± 0.9	0.6 ± 0.4
terran_20_vs_20	3.7 ± 0.4	8.8 ± 1.5	4.3 ± 0.6	6.0 ± 1.6	5.2 ± 1.2
terran_20_vs_23	0.5 ± 0.3	1.8 ± 1.4	0.6 ± 0.6	0.3 ± 0.4	0.1 ± 0.1
zerg_5_vs_5	5.6 ± 1.3	7.9 ± 1.0	5.3 ± 1.1	4.7 ± 1.6	3.8 ± 0.6
zerg_10_vs_10	3.4 ± 1.7	8.9 ± 2.1	5.6 ± 1.5	6.7 ± 1.1	3.5 ± 1.0
zerg_10_vs_11	2.5 ± 1.0	6.8 ± 1.1	4.6 ± 1.6	4.5 ± 1.3	2.8 ± 0.6
zerg_20_vs_20	0.3 ± 0.3	2.4 ± 0.6	0.3 ± 0.3	0.4 ± 0.5	0.3 ± 0.3
zerg_20_vs_23	0.6 ± 0.3	2.7 ± 0.4	1.4 ± 0.7	1.1 ± 1.0	1.2 ± 0.7

In [133...

```
show_box_chart(pd_returns_llm, PROTOSS_ENVS, tag="returns")
show_box_chart(pd_returns_llm, TERRAN_ENVS, tag="returns")
show_box_chart(pd_returns_llm, ZERG_ENVS, tag="returns")
```



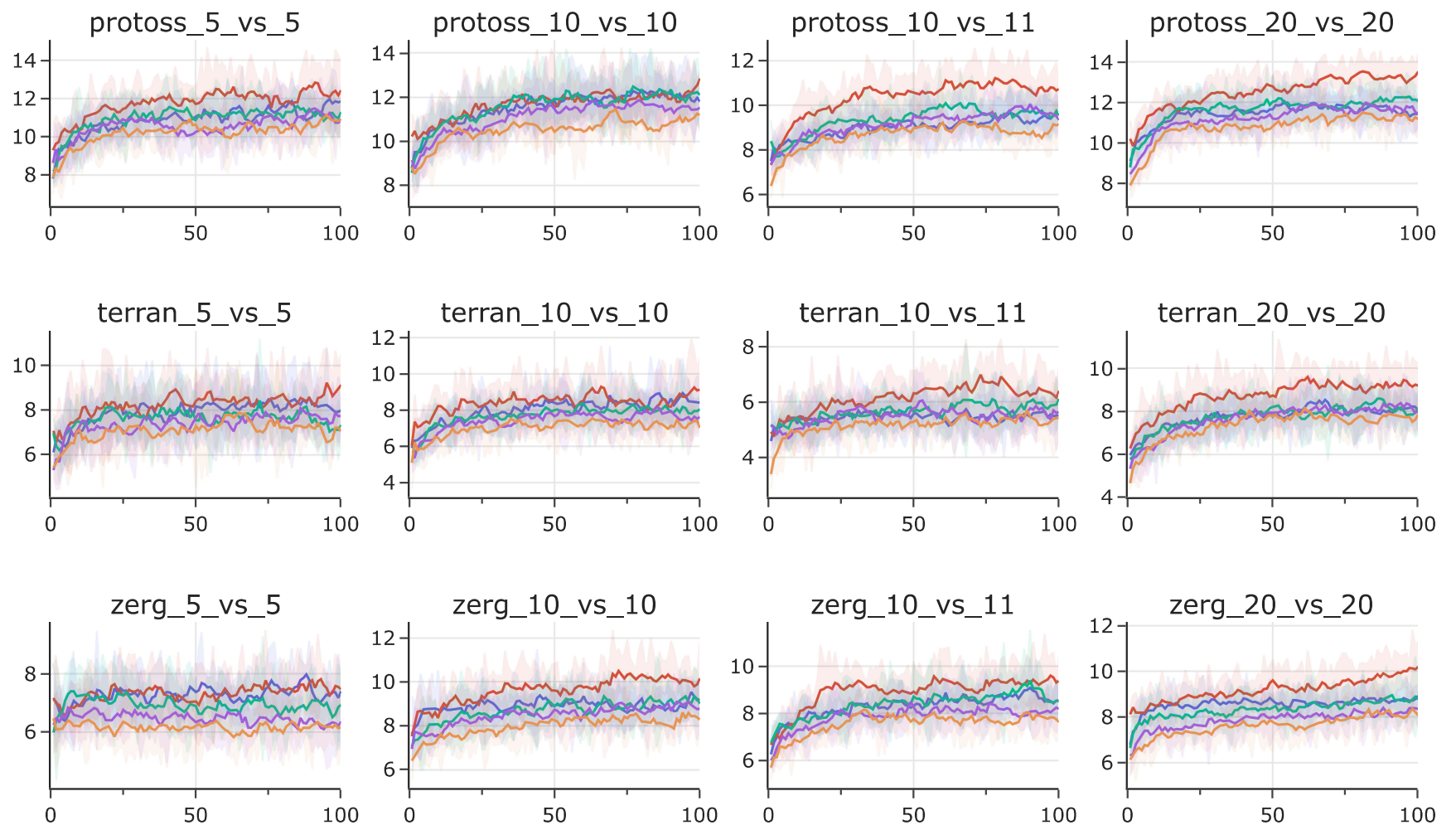
```
In [134... show_box_chart(pd_winrates_llm, PROTOSS_ENVS, tag="winrates")
show_box_chart(pd_winrates_llm, TERRAN_ENVS, tag="winrates")
show_box_chart(pd_winrates_llm, ZERG_ENVS, tag="winrates")
```



```
In [135... plotly_figs = {}
for env_name in SMACV2_ENV_NAMES:
    plotly_figs[env_name] = create_scatters(env_name, ablation_exsize_returns_llm, algos=[50, 200, 400, 800, 1200])

show_fig(plotly_figs, SMACV2_ENV_NAMES, n_rows=3, n_cols=5, height=640, width=1200, title="Ablation: SMACv2 - Returns - Rule-b
```

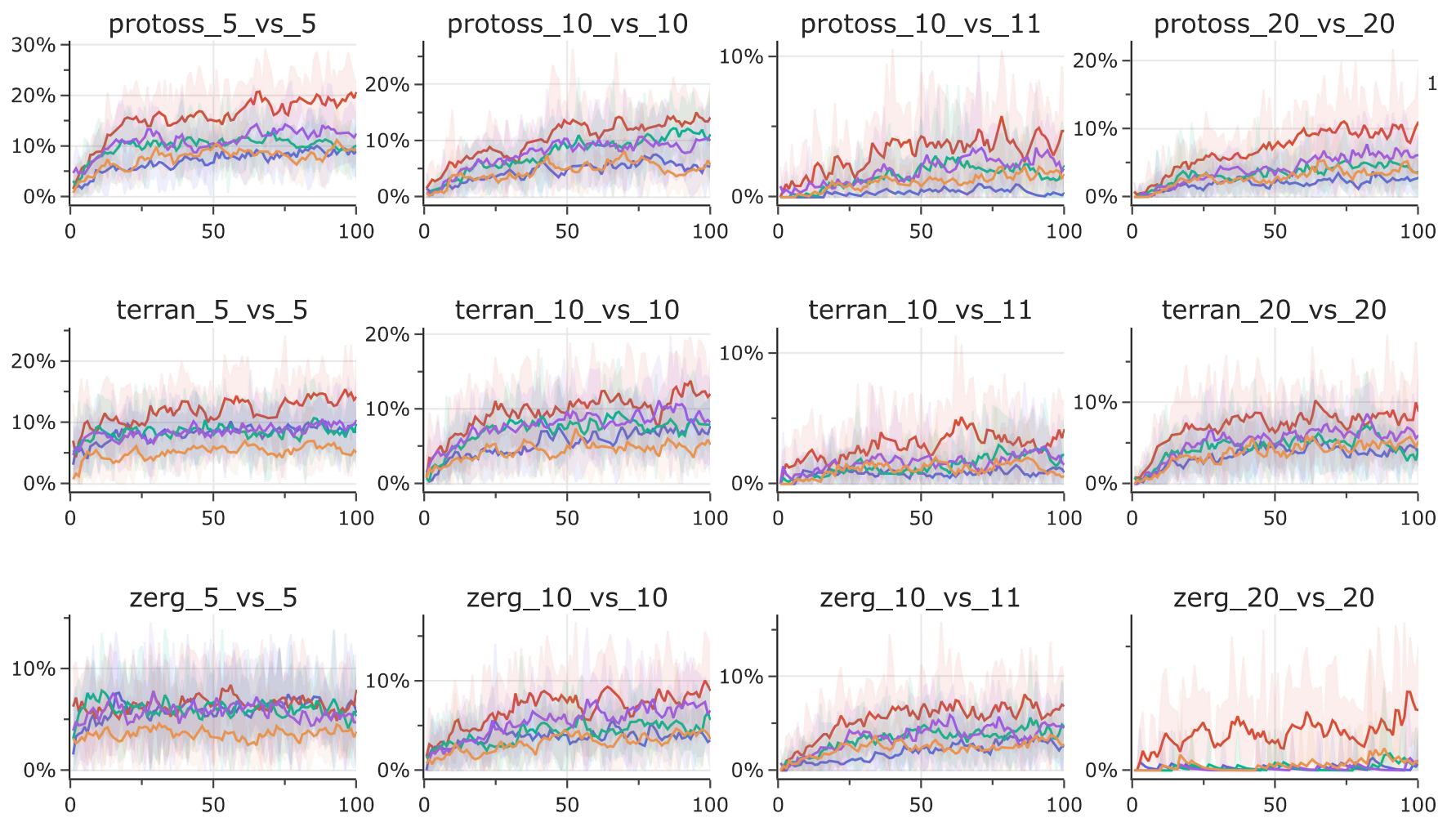
Ablation: SMACv2 - Returns - Rule-based



Saving Ablation_SMACv2_Returns_Rule-based.pdf

```
In [136... plotly_figs = {}  
for env_name in SMACV2_ENV_NAMES:  
    plotly_figs[env_name] = create_scatters(env_name, ablation_exsize_winrates_llm, algos=[50, 200, 400, 800, 1200], tag="winr  
show_fig(plotly_figs, SMACV2_ENV_NAMES, n_rows=3, n_cols=5, height=640, width=1200, title="Ablation: SMACv2 - Winrates - Rule-
```

Ablation: SMACv2 - Winrates - Rule-based



Saving Ablation_SMACv2_Winrates_Rule-based.pdf

In [137...

RENAME_ALGOS_GPT = {
 "miso-mini": "MisoDICE (gpt-4o-mini)",
 "miso_alpha0.05": "MisoDICE (gpt-4o)",
}

In [138...

data_returns_llm, data_winrates_llm = load_data(use_llm=True, algos=["miso-mini", "miso_alpha0.05"])
pd_returns_llm = pd.DataFrame.from_dict(data_returns_llm).rename(columns=RENAME_ALGOS_GPT)
pd_winrates_llm = pd.DataFrame.from_dict(data_winrates_llm).rename(columns=RENAME_ALGOS_GPT)

In [139...

pd_returns_llm.map(analyze_data)

Out[139...

	MisoDICE (gpt-4o-mini)	MisoDICE (gpt-4o)
2c_vs_64zg	12.2 ± 0.6	16.4 ± 1.3
5m_vs_6m	6.7 ± 0.9	7.3 ± 0.1
6h_vs_8z	8.1 ± 0.2	8.7 ± 0.2
corridor	3.8 ± 0.8	5.8 ± 0.8
protoss_5_vs_5	12.0 ± 0.3	12.4 ± 0.5
protoss_10_vs_10	12.0 ± 0.5	12.9 ± 0.2
protoss_10_vs_11	10.6 ± 0.3	10.7 ± 0.4
protoss_20_vs_20	12.4 ± 0.5	13.5 ± 0.5
protoss_20_vs_23	10.3 ± 0.1	10.6 ± 0.2
terran_5_vs_5	8.1 ± 0.5	9.1 ± 0.3
terran_10_vs_10	8.6 ± 0.9	9.1 ± 1.3
terran_10_vs_11	6.0 ± 0.3	6.4 ± 0.5
terran_20_vs_20	8.1 ± 0.5	9.2 ± 0.6
terran_20_vs_23	5.3 ± 0.3	5.6 ± 0.4
zerg_5_vs_5	7.0 ± 0.4	7.5 ± 0.1
zerg_10_vs_10	9.6 ± 0.2	10.2 ± 0.6
zerg_10_vs_11	9.2 ± 0.6	9.4 ± 0.3
zerg_20_vs_20	8.9 ± 0.4	10.2 ± 0.6
zerg_20_vs_23	9.0 ± 0.2	9.5 ± 0.2

In [140...

pd_winrates_llm.map(lambda x: analyze_data(x, tag="winrates"))

Out[140...

	MisoDICE (gpt-4o-mini)	MisoDICE (gpt-4o)
2c_vs_64zg	2.1 ± 1.1	13.0 ± 9.0
5m_vs_6m	1.2 ± 0.4	1.2 ± 0.5
6h_vs_8z	0.1 ± 0.1	1.1 ± 0.8
corridor	0.6 ± 0.7	1.4 ± 0.6
protoss_5_vs_5	12.6 ± 2.6	20.7 ± 0.9
protoss_10_vs_10	10.5 ± 3.5	14.1 ± 2.1
protoss_10_vs_11	4.0 ± 1.8	4.7 ± 0.3
protoss_20_vs_20	6.0 ± 0.9	11.0 ± 3.2
protoss_20_vs_23	2.3 ± 1.4	3.8 ± 2.0
terran_5_vs_5	10.0 ± 1.4	14.2 ± 3.1
terran_10_vs_10	9.2 ± 2.1	12.0 ± 1.7
terran_10_vs_11	2.2 ± 1.1	4.2 ± 1.6
terran_20_vs_20	6.1 ± 2.1	8.8 ± 1.5
terran_20_vs_23	0.9 ± 0.6	1.8 ± 1.4
zerg_5_vs_5	5.3 ± 1.1	7.9 ± 1.0
zerg_10_vs_10	7.1 ± 0.9	8.9 ± 2.1
zerg_10_vs_11	5.0 ± 0.9	6.8 ± 1.1
zerg_20_vs_20	0.8 ± 0.8	2.4 ± 0.6
zerg_20_vs_23	1.7 ± 0.8	2.7 ± 0.4

```
In [141... RENAME_ALGOS_2 = {
    "miso_alpha0.00": "MisoDICE ( $\alpha=0.00$ )",
    "miso_alpha0.05": "MisoDICE ( $\alpha=0.05$ )",
    "miso_alpha0.10": "MisoDICE ( $\alpha=0.10$ )",
    "miso_alpha0.50": "MisoDICE ( $\alpha=0.50$ )",
    "miso_alpha1.00": "MisoDICE ( $\alpha=1.00$ )",
    "miso_alpha10.00": "MisoDICE ( $\alpha=10.0$ )",
}
data_returns_llm, data_winrates_llm = load_data(use_llm=True, algos=ABLATION_ALGOS)
pd_returns_llm = pd.DataFrame.from_dict(data_returns_llm).rename(columns=RENAME_ALGOS_2)
pd_winrates_llm = pd.DataFrame.from_dict(data_winrates_llm).rename(columns=RENAME_ALGOS_2)
```

```
In [142... pd_returns_llm.map(analyze_data)
```

Out[142...

	MisoDICE ($\alpha=0.00$)	MisoDICE ($\alpha=0.05$)	MisoDICE ($\alpha=0.10$)	MisoDICE ($\alpha=0.50$)	MisoDICE ($\alpha=1.00$)	MisoDICE ($\alpha=10.0$)
2c_vs_64zg	15.9 ± 1.0	16.4 ± 1.3	15.3 ± 1.1	14.9 ± 0.7	13.3 ± 0.9	10.9 ± 0.3
5m_vs_6m	6.6 ± 1.4	7.3 ± 0.1	6.3 ± 1.0	6.5 ± 0.5	6.9 ± 0.2	6.7 ± 0.1
6h_vs_8z	8.3 ± 0.2	8.7 ± 0.2	8.0 ± 0.2	8.0 ± 0.2	7.7 ± 0.2	7.6 ± 0.1
corridor	5.3 ± 0.6	5.8 ± 0.8	5.3 ± 0.6	5.1 ± 0.2	3.8 ± 0.9	1.7 ± 0.1
protoss_5_vs_5	12.4 ± 0.4	12.4 ± 0.5	11.7 ± 0.4	11.8 ± 0.4	11.9 ± 0.3	11.7 ± 0.4
protoss_10_vs_10	12.1 ± 0.2	12.9 ± 0.2	12.2 ± 0.6	12.1 ± 0.3	11.6 ± 0.5	12.1 ± 0.4
protoss_10_vs_11	10.7 ± 0.5	10.7 ± 0.4	10.1 ± 0.5	10.1 ± 0.4	9.8 ± 0.3	9.9 ± 0.1
protoss_20_vs_20	12.4 ± 0.2	13.5 ± 0.5	12.3 ± 0.4	12.2 ± 0.5	11.8 ± 0.2	12.1 ± 0.4
protoss_20_vs_23	10.4 ± 0.4	10.6 ± 0.2	10.1 ± 0.3	10.1 ± 0.2	10.0 ± 0.2	9.6 ± 0.5
terran_5_vs_5	8.8 ± 0.7	9.1 ± 0.3	8.4 ± 0.8	8.5 ± 0.2	7.9 ± 0.4	8.2 ± 0.3
terran_10_vs_10	8.9 ± 0.8	9.1 ± 1.3	8.5 ± 0.6	8.6 ± 0.9	7.8 ± 0.7	8.1 ± 0.7
terran_10_vs_11	5.9 ± 0.4	6.4 ± 0.5	5.8 ± 0.3	5.9 ± 0.4	5.7 ± 0.5	5.8 ± 0.6
terran_20_vs_20	9.0 ± 0.6	9.2 ± 0.6	8.5 ± 0.7	8.4 ± 0.6	8.2 ± 0.6	8.4 ± 0.7
terran_20_vs_23	5.6 ± 0.2	5.6 ± 0.4	5.3 ± 0.3	5.1 ± 0.3	5.1 ± 0.3	5.1 ± 0.1
zerg_5_vs_5	7.3 ± 0.5	7.5 ± 0.1	7.2 ± 0.6	6.8 ± 0.2	6.4 ± 0.3	6.4 ± 0.3
zerg_10_vs_10	9.7 ± 0.4	10.2 ± 0.6	9.2 ± 0.3	9.2 ± 0.4	8.7 ± 0.2	8.9 ± 0.4
zerg_10_vs_11	9.0 ± 0.5	9.4 ± 0.3	8.5 ± 0.2	8.8 ± 0.3	8.7 ± 0.4	8.3 ± 0.5
zerg_20_vs_20	9.0 ± 0.4	10.2 ± 0.6	9.0 ± 0.3	9.1 ± 0.7	8.8 ± 0.4	8.9 ± 0.4
zerg_20_vs_23	8.9 ± 0.2	9.5 ± 0.2	8.8 ± 0.4	8.9 ± 0.3	8.5 ± 0.4	8.6 ± 0.4

```
In [143... pd_winrates_llm.map(lambda x: analyze_data(x, tag="winrates"))
```

Out[143...

	MisoDICE ($\alpha=0.00$)	MisoDICE ($\alpha=0.05$)	MisoDICE ($\alpha=0.10$)	MisoDICE ($\alpha=0.50$)	MisoDICE ($\alpha=1.00$)	MisoDICE ($\alpha=10.0$)
2c_vs_64zg	11.0 \pm 5.8	13.0 \pm 9.0	10.0 \pm 5.0	9.3 \pm 2.9	7.1 \pm 4.1	1.7 \pm 0.9
5m_vs_6m	1.1 \pm 1.0	1.2 \pm 0.5	0.6 \pm 0.4	0.7 \pm 0.4	0.9 \pm 0.7	0.7 \pm 0.5
6h_vs_8z	1.0 \pm 0.8	1.1 \pm 0.8	1.1 \pm 0.8	0.8 \pm 0.5	0.4 \pm 0.3	1.0 \pm 0.5
corridor	0.4 \pm 0.2	1.4 \pm 0.6	1.2 \pm 0.5	0.5 \pm 0.4	1.4 \pm 0.7	0.9 \pm 0.4
protoss_5_vs_5	12.1 \pm 1.6	20.7 \pm 0.9	14.4 \pm 2.1	12.9 \pm 2.0	14.1 \pm 2.2	17.2 \pm 2.5
protoss_10_vs_10	8.4 \pm 3.3	14.1 \pm 2.1	9.9 \pm 2.2	9.5 \pm 2.6	9.0 \pm 1.7	11.1 \pm 2.3
protoss_10_vs_11	3.3 \pm 1.1	4.7 \pm 0.3	4.3 \pm 0.4	3.5 \pm 1.3	3.5 \pm 1.1	4.3 \pm 1.2
protoss_20_vs_20	5.4 \pm 0.4	11.0 \pm 3.2	5.5 \pm 1.1	6.4 \pm 1.0	6.4 \pm 0.7	7.8 \pm 2.4
protoss_20_vs_23	4.3 \pm 0.6	3.8 \pm 2.0	2.3 \pm 1.4	3.5 \pm 1.6	3.3 \pm 0.8	2.5 \pm 1.3
terran_5_vs_5	12.4 \pm 2.9	14.2 \pm 3.1	11.2 \pm 0.8	12.1 \pm 1.6	12.0 \pm 3.2	12.2 \pm 3.2
terran_10_vs_10	9.3 \pm 1.2	12.0 \pm 1.7	8.6 \pm 1.9	9.2 \pm 3.6	8.8 \pm 1.9	9.3 \pm 3.2
terran_10_vs_11	2.3 \pm 0.4	4.2 \pm 1.6	3.5 \pm 1.2	2.3 \pm 0.7	2.2 \pm 0.8	3.6 \pm 1.5
terran_20_vs_20	4.8 \pm 1.6	8.8 \pm 1.5	4.9 \pm 1.5	5.9 \pm 2.5	4.9 \pm 1.9	9.3 \pm 2.1
terran_20_vs_23	1.5 \pm 0.7	1.8 \pm 1.4	1.2 \pm 0.7	1.8 \pm 0.7	1.2 \pm 0.9	1.7 \pm 1.0
zerg_5_vs_5	7.1 \pm 1.6	7.9 \pm 1.0	7.4 \pm 1.0	5.9 \pm 0.7	6.4 \pm 2.7	7.5 \pm 2.1
zerg_10_vs_10	6.7 \pm 1.2	8.9 \pm 2.1	7.2 \pm 2.6	7.0 \pm 0.8	7.1 \pm 0.9	6.9 \pm 1.0
zerg_10_vs_11	5.0 \pm 1.4	6.8 \pm 1.1	5.0 \pm 1.5	4.1 \pm 1.9	5.0 \pm 1.2	4.5 \pm 1.4
zerg_20_vs_20	1.3 \pm 0.5	2.4 \pm 0.6	1.2 \pm 0.4	1.7 \pm 2.1	1.6 \pm 0.3	2.6 \pm 0.6
zerg_20_vs_23	1.4 \pm 0.5	2.7 \pm 0.4	1.0 \pm 0.4	2.5 \pm 0.7	2.7 \pm 1.7	2.1 \pm 0.7

In [144...

```
MAIN_ALGOS_TMP = ["bc_beta0.0", "bc_beta0.5", "bc_beta1.0", "indd", "vdn", "miso_alpha0.05"]
```

In [145...

```
def load_data_tmp(use_llm=False, exsize=200, algos=None):
    data_returns = defaultdict(lambda: defaultdict(lambda: defaultdict(list)))
    data_winrates = defaultdict(lambda: defaultdict(lambda: defaultdict(list)))
    if not algos:
        algos = ALGOS
    for algo in algos:
        for env_name in MAMUJOCO_ENV_NAMES + SMACV1_ENV_NAMES + SMACV2_ENV_NAMES:
            for seed in range(4):
                if use_llm:
                    if env_name in MAMUJOCO_ENV_NAMES:
                        continue
                    path = f"logs_ablation/{algo}/{env_name}_llm/seed{seed}/exsize{exsize}/results.json"
                else:
                    path = f"logs_ablation/{algo}/{env_name}/seed{seed}/exsize{exsize}/results.json"

                if algo == "omap1":
                    path = f"logs_ablation/{algo}/{env_name}_llm/seed{seed}/results.json"

                data = load_results(path)
                for step in range(101):
                    if step != 100:
                        continue
                    result = data[f"step_{step}"] if algo != "omap1" else data
                    if "returns" in result:
                        data_returns[algo][env_name][step].append(np.mean(result["returns"]))
                    if "winrates" in result:
                        data_winrates[algo][env_name][step].append(np.mean(result["winrates"]))
    return data_returns, data_winrates
```

In [146...

```
data_returns_llm, data_winrates_llm = load_data_tmp(use_llm=True, algos=MAIN_ALGOS_TMP)
pd_returns_llm = pd.DataFrame.from_dict(data_returns_llm).rename(columns=RENAME_ALGOS)
pd_winrates_llm = pd.DataFrame.from_dict(data_winrates_llm).rename(columns=RENAME_ALGOS)
```

In [147...

```
pd_returns_llm.map(analyze_data)
```

Out[147...

	BC ($\beta=0.0$)	BC ($\beta=0.5$)	BC ($\beta=1.0$)	INDD	VDN	MisoDICE (ours)
2c_vs_64zg	11.7 \pm 0.4	11.3 \pm 0.1	13.1 \pm 0.8	15.1 \pm 1.3	15.9 \pm 2.0	16.1 \pm 1.8
5m_vs_6m	6.1 \pm 1.4	6.8 \pm 1.4	7.2 \pm 0.2	7.4 \pm 0.2	7.4 \pm 0.3	7.4 \pm 0.1
6h_vs_8z	8.4 \pm 0.1	8.4 \pm 0.2	8.3 \pm 0.3	8.2 \pm 0.3	8.5 \pm 0.1	8.9 \pm 0.1
corridor	2.1 \pm 0.3	1.9 \pm 0.3	5.7 \pm 1.5	2.0 \pm 0.2	5.8 \pm 1.5	6.1 \pm 1.6
protoss_5_vs_5	13.3 \pm 0.1	12.6 \pm 2.8	12.0 \pm 1.2	11.9 \pm 1.9	13.3 \pm 0.5	13.7 \pm 1.2
protoss_10_vs_10	13.1 \pm 0.9	12.5 \pm 1.4	12.8 \pm 0.9	12.3 \pm 0.8	13.2 \pm 0.5	14.0 \pm 1.2
protoss_10_vs_11	10.8 \pm 1.3	10.6 \pm 0.9	10.9 \pm 0.8	10.1 \pm 1.1	11.4 \pm 0.5	12.0 \pm 1.3
protoss_20_vs_20	12.4 \pm 1.4	12.2 \pm 0.5	13.0 \pm 0.4	12.7 \pm 0.6	13.3 \pm 1.4	13.9 \pm 1.6
protoss_20_vs_23	10.1 \pm 1.5	9.5 \pm 0.5	10.2 \pm 0.8	10.1 \pm 0.9	10.4 \pm 0.8	10.5 \pm 1.1
terran_5_vs_5	8.9 \pm 0.9	8.4 \pm 1.0	8.7 \pm 0.9	8.9 \pm 1.8	9.4 \pm 2.3	9.4 \pm 1.5
terran_10_vs_10	7.5 \pm 1.0	8.2 \pm 1.0	7.7 \pm 1.2	7.9 \pm 0.7	8.9 \pm 1.3	9.0 \pm 0.6
terran_10_vs_11	6.3 \pm 0.4	5.9 \pm 1.4	5.8 \pm 0.4	5.9 \pm 0.7	7.1 \pm 1.2	7.3 \pm 1.0
terran_20_vs_20	8.6 \pm 1.2	8.3 \pm 0.8	7.9 \pm 0.2	8.3 \pm 1.0	8.7 \pm 1.0	9.3 \pm 1.2
terran_20_vs_23	5.1 \pm 0.2	5.5 \pm 0.5	5.4 \pm 0.3	4.8 \pm 0.4	5.6 \pm 0.5	5.9 \pm 0.7
zerg_5_vs_5	7.8 \pm 1.2	7.0 \pm 0.9	7.1 \pm 1.4	6.8 \pm 0.3	7.8 \pm 0.8	8.0 \pm 0.6
zerg_10_vs_10	8.7 \pm 1.2	8.2 \pm 0.5	9.2 \pm 0.8	9.6 \pm 1.2	9.6 \pm 1.7	9.9 \pm 0.5
zerg_10_vs_11	9.6 \pm 0.4	9.3 \pm 1.5	8.8 \pm 1.2	9.3 \pm 1.0	9.8 \pm 0.9	9.9 \pm 0.9
zerg_20_vs_20	9.2 \pm 0.8	9.3 \pm 0.6	9.4 \pm 1.1	9.2 \pm 0.7	9.5 \pm 0.8	11.1 \pm 1.3
zerg_20_vs_23	9.2 \pm 1.1	8.9 \pm 0.5	8.2 \pm 0.6	9.3 \pm 1.0	9.5 \pm 0.3	9.5 \pm 0.1

In [148...

```
pd_winrates_llm.map(lambda x: analyze_data(x, tag="winrates"))
```

Out[148...

	BC ($\beta=0.0$)	BC ($\beta=0.5$)	BC ($\beta=1.0$)	INDD	VDN	MisoDICE (ours)
2c_vs_64zg	1.6 \pm 1.6	0.0 \pm 0.0	5.5 \pm 2.6	9.4 \pm 2.2	10.2 \pm 7.1	11.7 \pm 8.1
5m_vs_6m	0.8 \pm 1.4	1.6 \pm 1.6	0.8 \pm 1.4	0.8 \pm 1.4	1.6 \pm 1.6	1.6 \pm 1.6
6h_vs_8z	0.0 \pm 0.0	0.8 \pm 1.4	0.0 \pm 0.0	0.0 \pm 0.0	1.6 \pm 1.6	2.3 \pm 2.6
corridor	0.8 \pm 1.4	0.8 \pm 1.4	0.8 \pm 1.4	0.8 \pm 1.4	1.6 \pm 2.7	2.3 \pm 2.6
protoss_5_vs_5	13.3 \pm 3.4	17.2 \pm 8.4	20.3 \pm 4.7	17.2 \pm 3.5	20.3 \pm 6.8	21.1 \pm 11.6
protoss_10_vs_10	10.2 \pm 3.4	10.2 \pm 3.4	8.6 \pm 6.0	6.2 \pm 2.2	10.2 \pm 2.6	15.6 \pm 4.4
protoss_10_vs_11	1.6 \pm 1.6	3.9 \pm 1.4	3.1 \pm 3.1	4.7 \pm 3.5	4.7 \pm 1.6	6.2 \pm 2.2
protoss_20_vs_20	8.6 \pm 6.4	2.3 \pm 2.6	9.4 \pm 2.2	7.0 \pm 7.8	9.4 \pm 2.2	10.9 \pm 5.2
protoss_20_vs_23	2.3 \pm 2.6	1.6 \pm 1.6	1.6 \pm 1.6	2.3 \pm 1.4	3.1 \pm 3.8	3.9 \pm 5.1
terran_5_vs_5	10.9 \pm 3.5	7.0 \pm 4.1	9.4 \pm 3.8	11.7 \pm 4.1	16.4 \pm 7.5	17.2 \pm 8.4
terran_10_vs_10	8.6 \pm 2.6	7.0 \pm 4.1	7.8 \pm 3.5	5.5 \pm 2.6	9.4 \pm 3.8	10.9 \pm 3.5
terran_10_vs_11	1.6 \pm 1.6	1.6 \pm 2.7	2.3 \pm 1.4	1.6 \pm 2.7	4.7 \pm 3.5	5.5 \pm 2.6
terran_20_vs_20	7.0 \pm 2.6	6.2 \pm 2.2	4.7 \pm 3.5	4.7 \pm 1.6	8.6 \pm 5.1	9.4 \pm 2.2
terran_20_vs_23	0.8 \pm 1.4	0.0 \pm 0.0	1.6 \pm 2.7	0.8 \pm 1.4	1.6 \pm 1.6	2.3 \pm 1.4
zerg_5_vs_5	6.2 \pm 2.2	5.5 \pm 1.4	3.9 \pm 2.6	6.2 \pm 3.8	7.0 \pm 4.6	7.8 \pm 4.7
zerg_10_vs_10	7.8 \pm 2.7	5.5 \pm 3.4	5.5 \pm 4.6	3.1 \pm 2.2	8.6 \pm 6.8	8.6 \pm 8.1
zerg_10_vs_11	3.9 \pm 1.4	4.7 \pm 3.5	0.8 \pm 1.4	3.9 \pm 2.6	8.6 \pm 3.4	8.6 \pm 4.6
zerg_20_vs_20	1.6 \pm 2.7	1.6 \pm 1.6	0.8 \pm 1.4	0.0 \pm 0.0	3.9 \pm 2.6	2.3 \pm 2.6
zerg_20_vs_23	0.8 \pm 1.4	1.6 \pm 1.6	1.6 \pm 1.6	0.0 \pm 0.0	1.6 \pm 2.7	1.6 \pm 2.7